

Business Intelligence and Big Data Analysis

Journal

By

**Zeeshan Ali Sayyad
Roll no: 004
MSc Cs Sem-2**



CERTIFICATE

This is here to certify that Mr. **Zeeshan Ali Sayyad**, Seat Number **004** M.Sc. Sem II Computer Science, has satisfactorily completed the required number of experiments prescribed by the Thakur College of Science and Commerce during the academic year 2021 – 2022.

Date:

Place: Mumbai

Teacher In-Charge

Head of Department

External Examiner

INDEX

| Sr. No. | Practical Name | Date |
|---------|--|---------------------------|
| 1 | Do data preprocessing on data obtained from databases imported from external sources. | 31 st Jan 2022 |
| 2 | Develop an application to implement defining subject areas, design of fact and dimension tables, data marts. | 14 th Jan 2022 |
| 3 | Develop an application to implement OLAP, roll-up, drill-down, slice, and dice operations. | 07 th Feb 2022 |
| 4 | Develop an application to construct a multidimensional data. | 14 th Feb 2022 |
| 5 | Design and create cube by identifying measures and dimensions for star schema. | 28 th Feb 2022 |
| 6 | Design and create cube by identifying measures and dimensions for snowflake schema. | 21 st Mar 2022 |
| 7 | Create and uses excel pivot table report based on data cube operations. | 04 th Apr 2022 |
| 8 | Prerequisites for Database applications. | 11 th Apr 2022 |

Practical 1

Aim: Do data preprocessing on data obtained from databases imported from external sources.

Theory:

What is data preprocessing?

Data preprocessing, a component of data preparation, describes any type of processing performed on raw data to prepare it for another data processing procedure. It has traditionally been an important preliminary step for the data mining process. More recently, data preprocessing techniques have been adapted for training machine learning models and AI models and for running inferences against them.

Data preprocessing transforms the data into a format that is more easily and effectively processed in data mining, machine learning and other data science tasks. The techniques are generally used at the earliest stages of the machine learning and AI development pipeline to ensure accurate results.

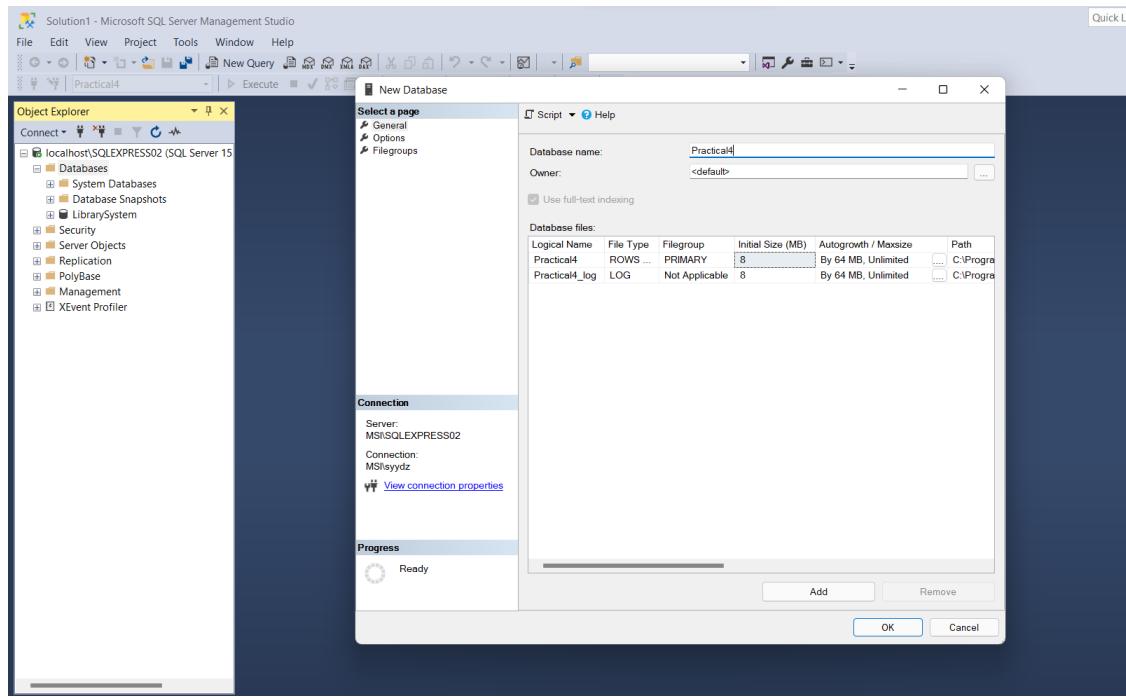
There are several different tools and methods used for preprocessing data, including the following:

- sampling, which selects a representative subset from a large population of data;
- transformation, which manipulates raw data to produce a single input;
- denoising, which removes noise from data;
- imputation, which synthesizes statistically relevant data for missing values;
- normalization, which organizes data for more efficient access; and
- feature extraction, which pulls out a relevant feature subset that is significant in a particular context.
- pulls out a relevant feature subset that is significant in a particular context.

#Steps:

1) Start Microsoft SQL server

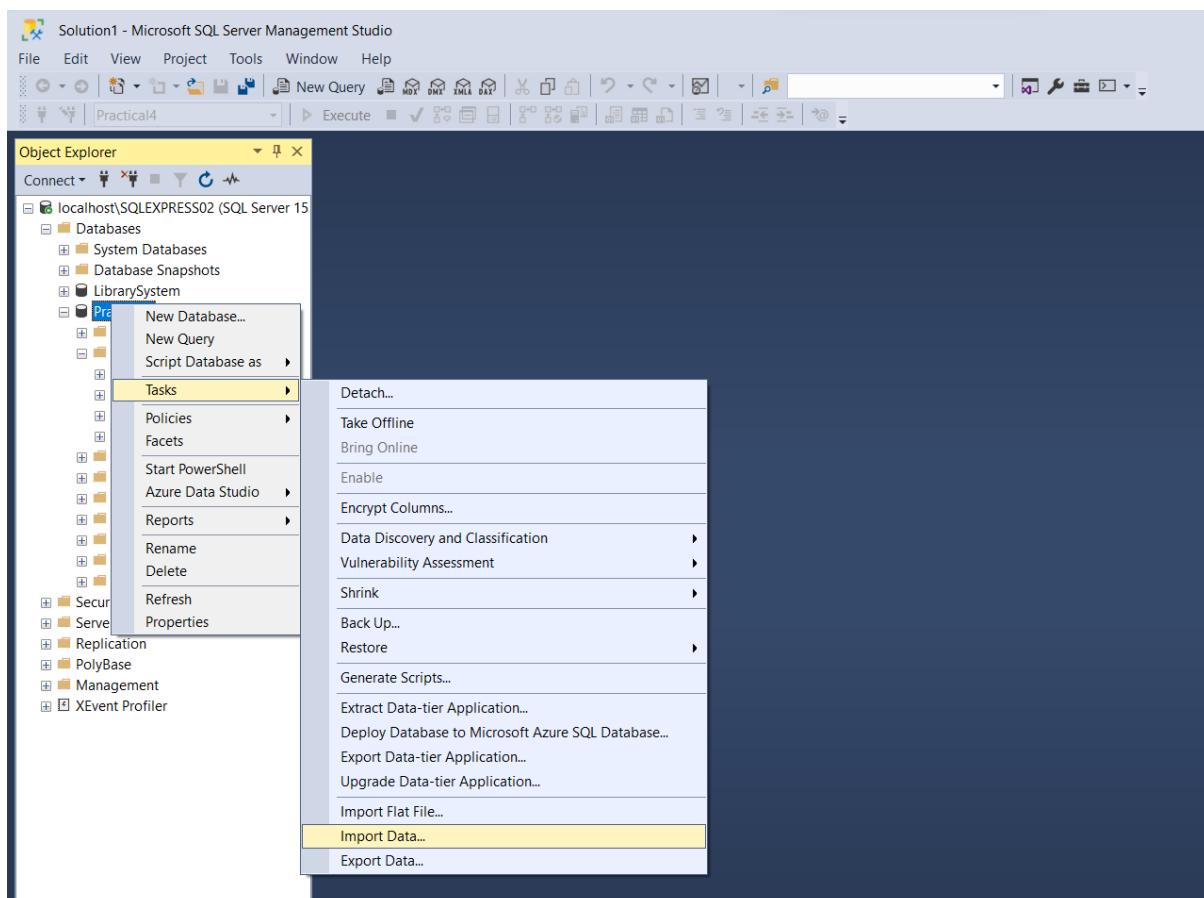
2) Right Click On Databases > New Database > Give Database Name > Click Ok

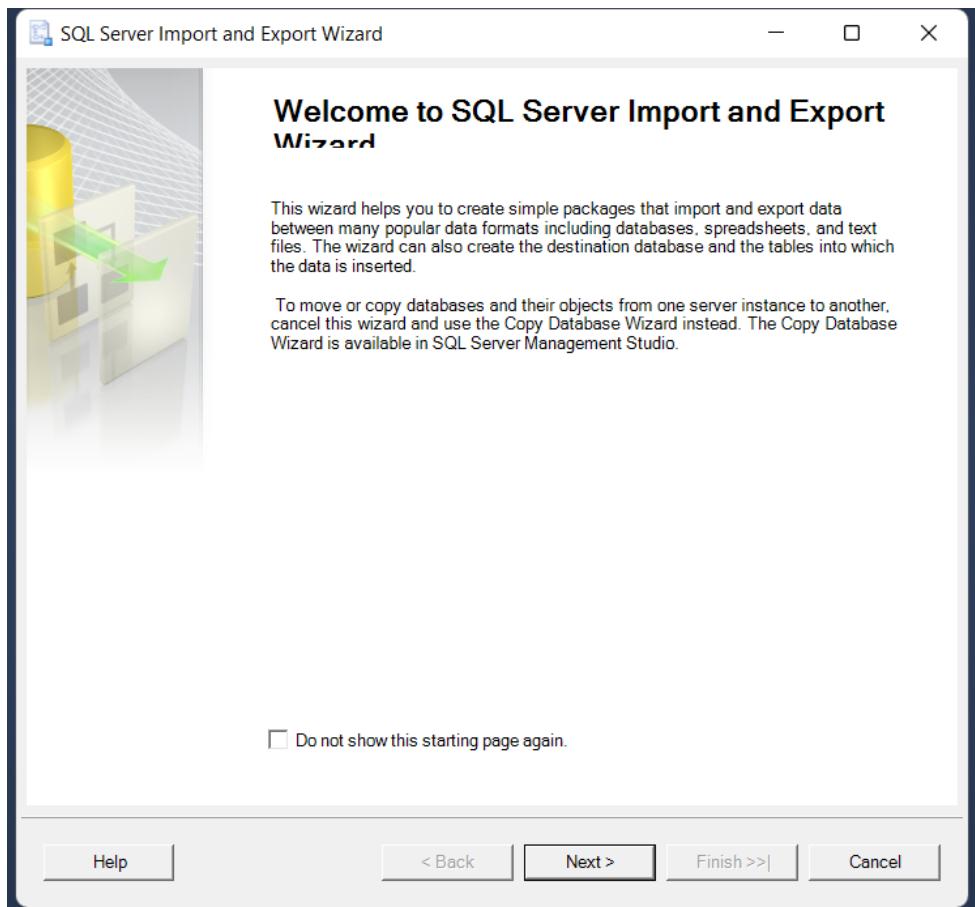


3) Create a new Microsoft Excel 97-2003 Worksheet (.xls) with columns Branch id, Branch name, Branch location and enter some data into rows.

| | A | B | C | D | E | F | G |
|---|------|--------|------------|---|---|---|---|
| 1 | B_id | B_name | B_location | | | | |
| 2 | 1 | Z1 | MUMBAI | | | | |
| 3 | 2 | Z2 | DUBAI | | | | |
| 4 | 3 | Z3 | Mexico | | | | |
| 5 | 4 | Z4 | New york | | | | |
| 6 | 5 | Z5 | Sweden | | | | |
| 7 | | | | | | | |

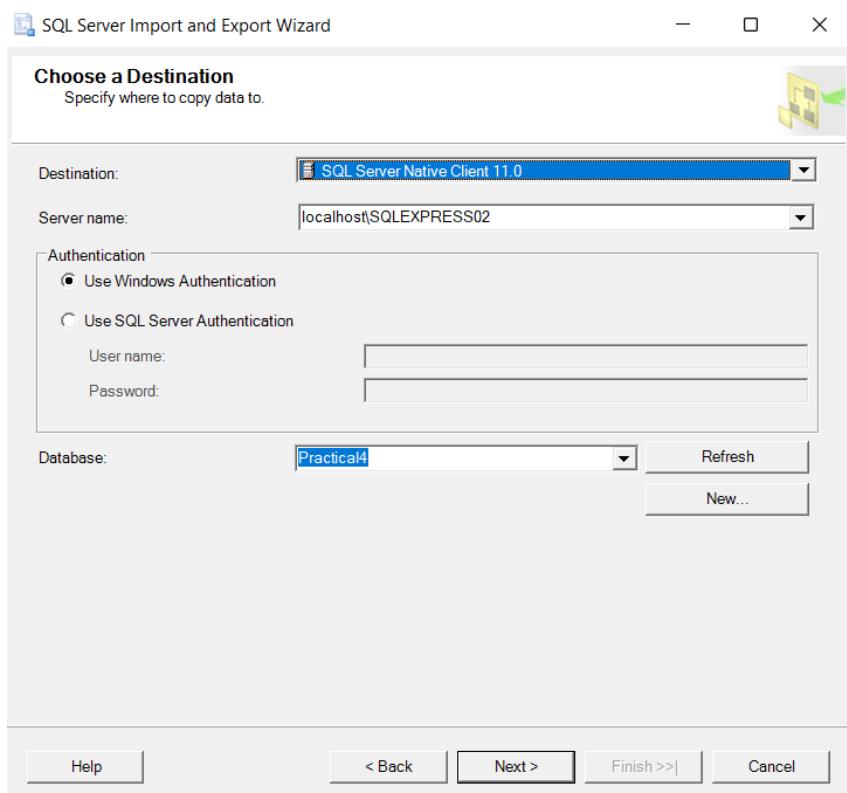
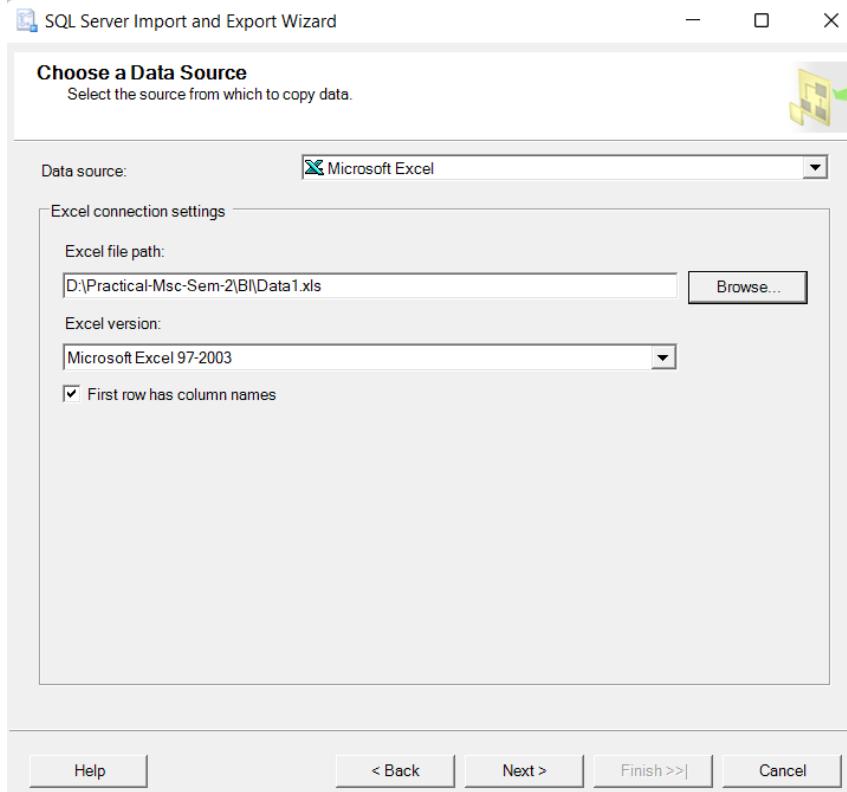
4) Right Click On Practical4 > Tasks > Select: Import Data

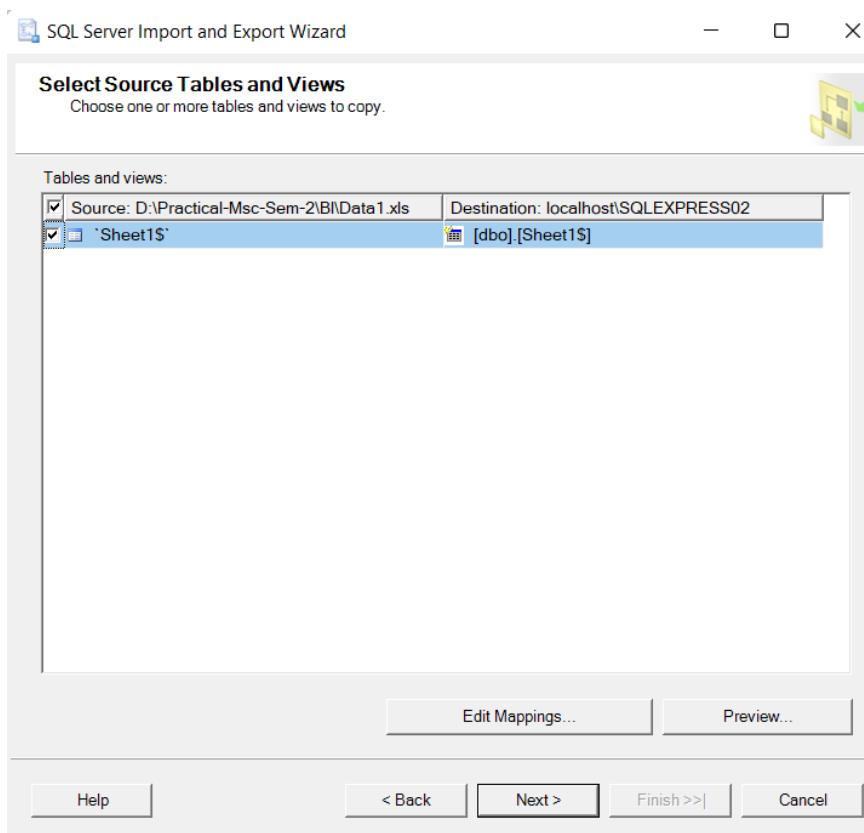
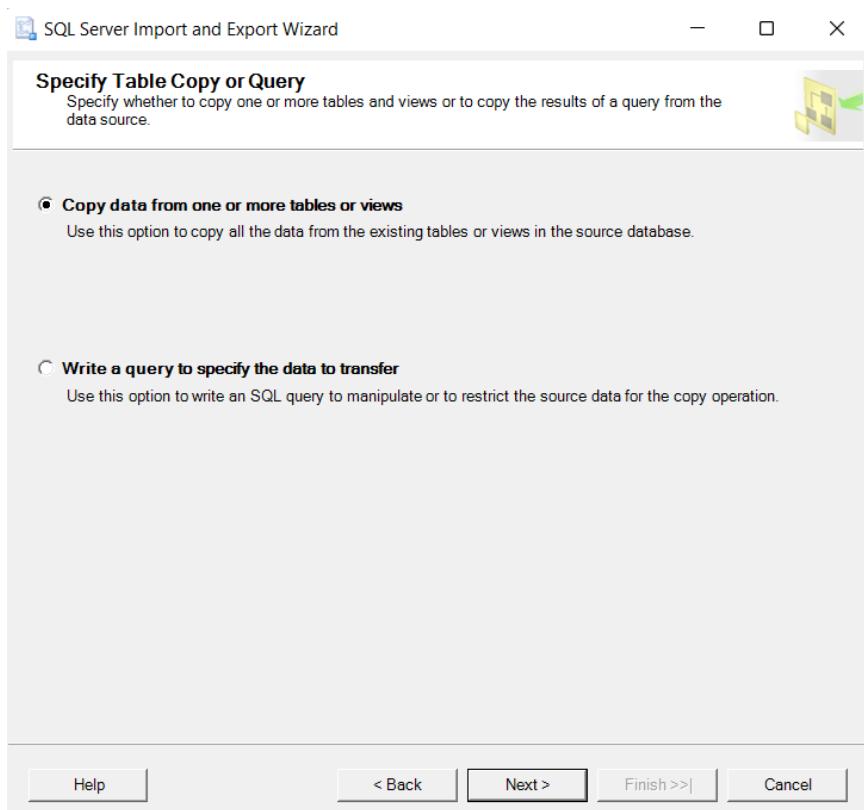


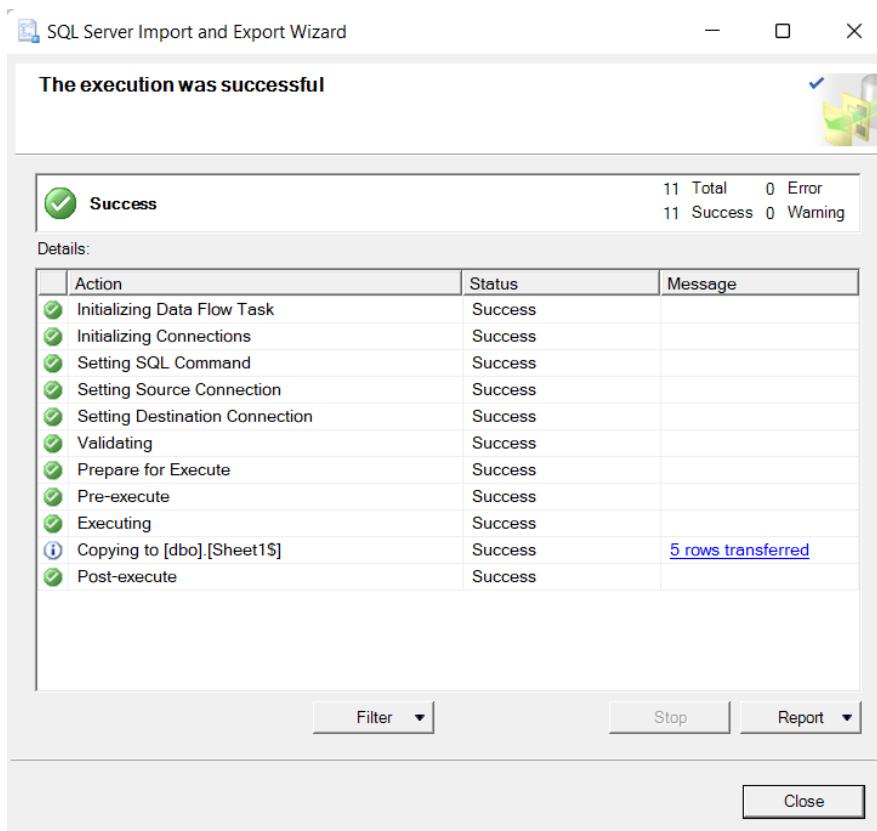
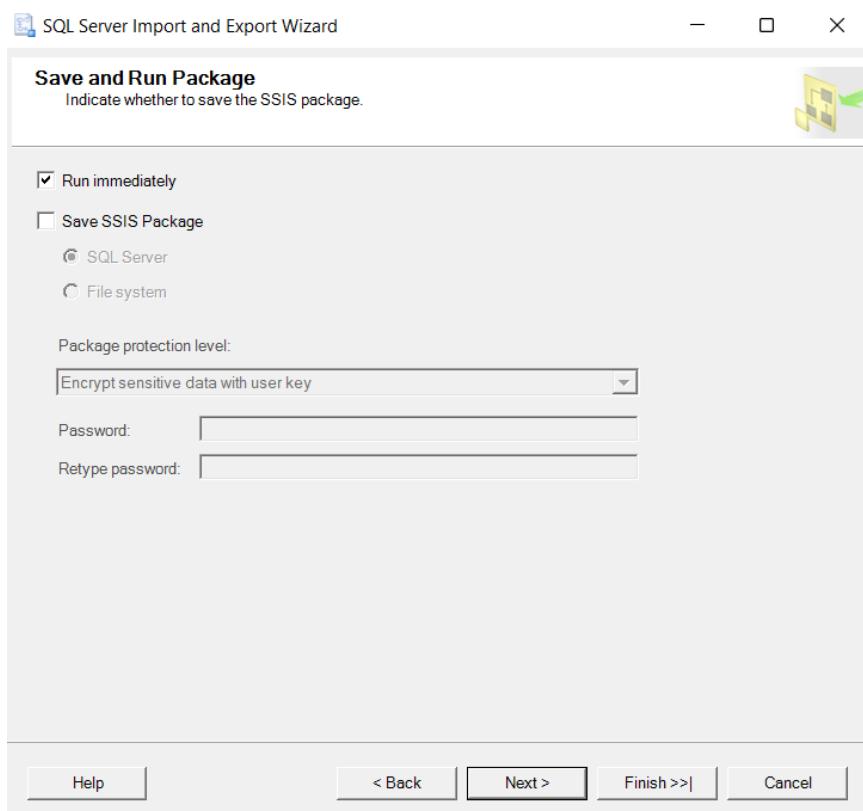


5) Click on Next > Select Data Source as: Microsoft Excel > Give Excel File Path > Select Destination as: SQL Server Native Client 11.0 > Click Next > Select Sheet 1 > Click Next > Click Finish > Close the window after successful execution.

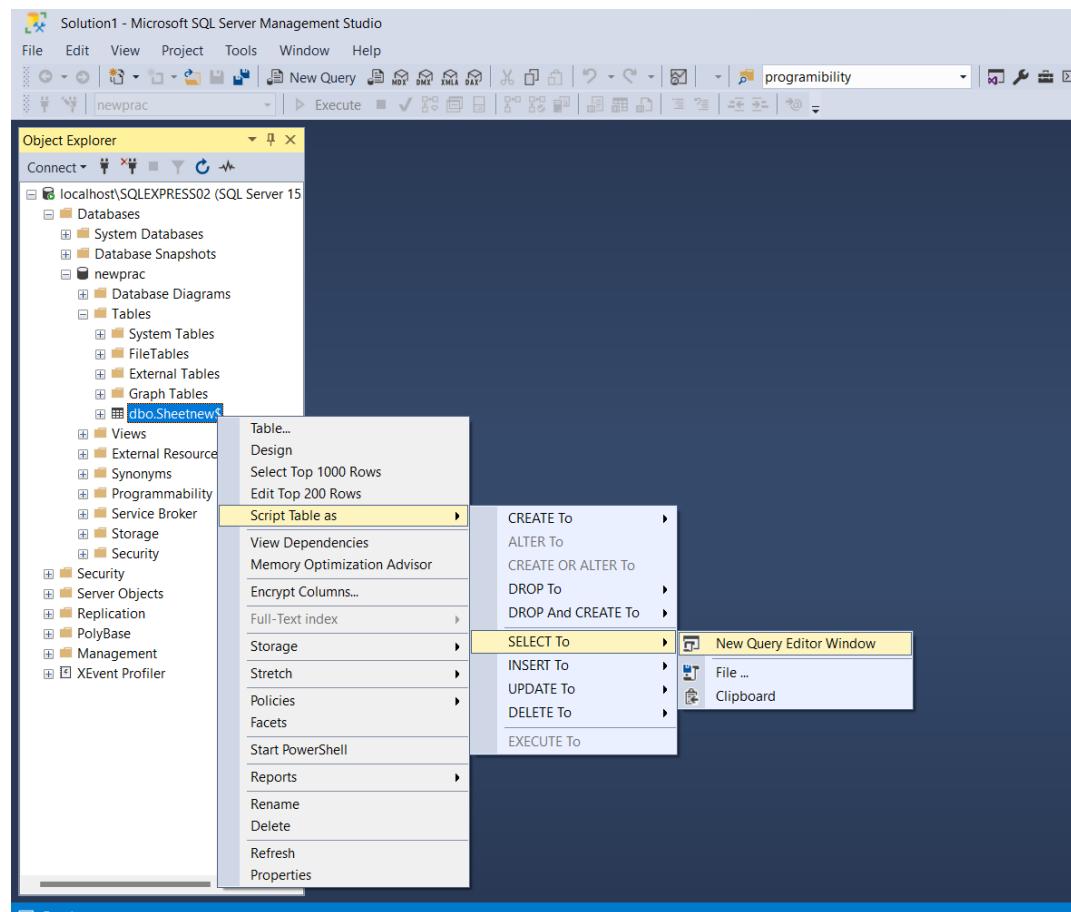
Note: Close the excel file before using it as Source File







6) Now expand the Tables Section > Right Click on dbo.Sheet\$1 > Script table as > Select To > Click on New Query Editor Window.



The screenshot shows the Microsoft SQL Server Management Studio interface with a new query window titled 'SQLQuery2.sql - localhost\SQLEXPRESS02\Practical4 (MSI\sydz (53))'. The query window displays the following T-SQL code:

```
USE [Practical4]
GO

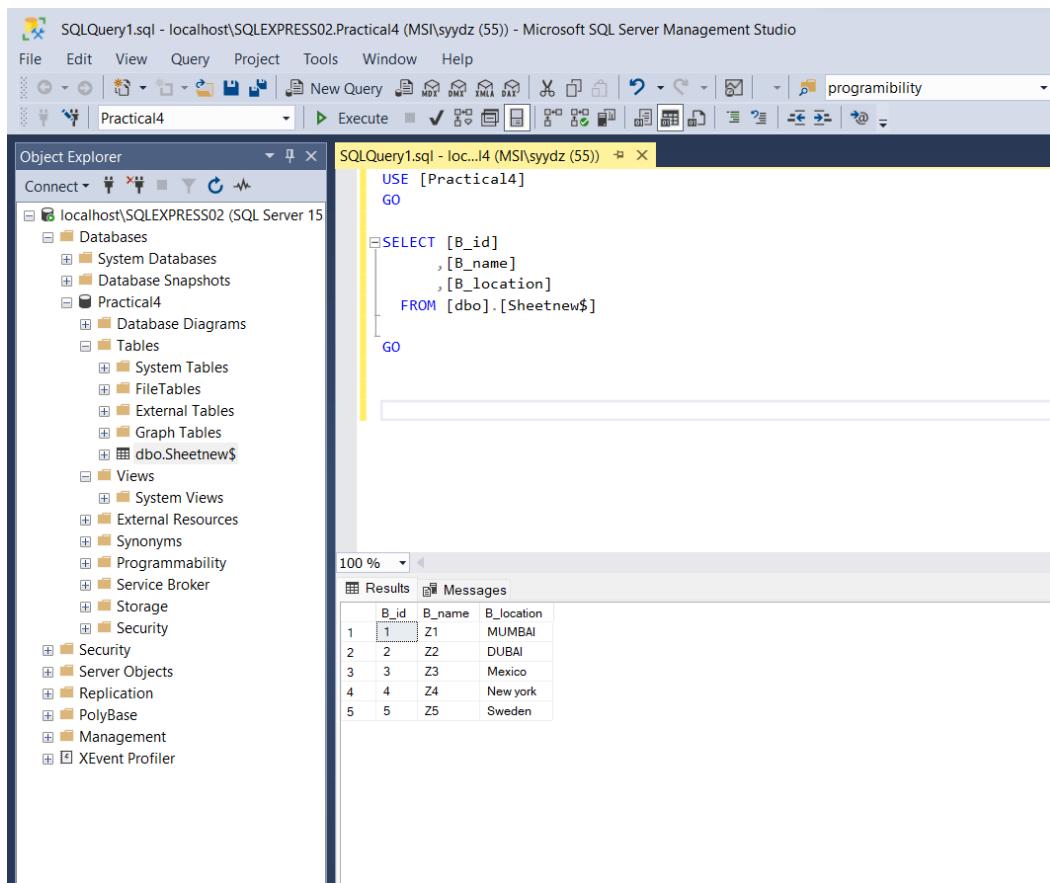
/****** Object: Table [dbo].[Sheet$1]  Script Date: 28-03-2022 00:04:58 *****/
SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER ON
GO

CREATE TABLE [dbo].[Sheet$1](
    [B_id] [float] NULL,
    [B_name] [nvarchar](255) NULL,
    [B_location] [nvarchar](255) NULL
) ON [PRIMARY]
GO
```

The Object Explorer on the left shows the same database structure as the previous screenshot, with 'dbo.Sheet\$1' selected. The status bar at the bottom indicates 'Connected. (1/1)' and the connection details 'localhost\SQLEXPRESS02 (15... MSI\sydz (53) Practical4 00:00:00 0 rows'.

7) Click on Execute



The screenshot shows the Microsoft SQL Server Management Studio interface. The Object Explorer on the left shows the database structure for 'localhost\SQLEXPRESS02'. A query window titled 'SQLQuery1.sql - localhost\SQLEXPRESS02.Practical4 (MSI\sydz (55)) - Microsoft SQL Server Management Studio' contains the following T-SQL code:

```
USE [Practical4]
GO

SELECT [B_id]
      ,[B_name]
      ,[B_location]
 FROM [dbo].[Sheetnew$]
GO
```

The results pane at the bottom displays a table with five rows of data:

| B_id | B_name | B_location |
|------|--------|------------|
| 1 | Z1 | MUMBAI |
| 2 | Z2 | DUBAI |
| 3 | Z3 | Mexico |
| 4 | Z4 | New York |
| 5 | Z5 | Sweden |

Conclusion: Data pre-processing on imported data from external sources implemented Successfully

Practical 2

Aim: Develop an application to implement defining subject areas, design of fact and dimension tables, data marts.

Theory:

Fact Table

A Fact Table is a central table in a star schema of a data warehouse. It is an important concept required for Data Warehousing and BI . A fact table stores quantitative information for analysis and is often denormalized. A fact table works with dimension tables and it holds the data to be analyzed and a dimension table stores data about the ways in which the data can be analyzed.

Thus, a fact table consists of two types of columns. The foreign keys column allows to join with dimension tables and the measure columns contain the data that is being analyzed.

A dimension table is a table in a star schema of a data warehouse. A dimension table stores attributes, or dimensions, that describe the objects in a fact table.

Dimension Table

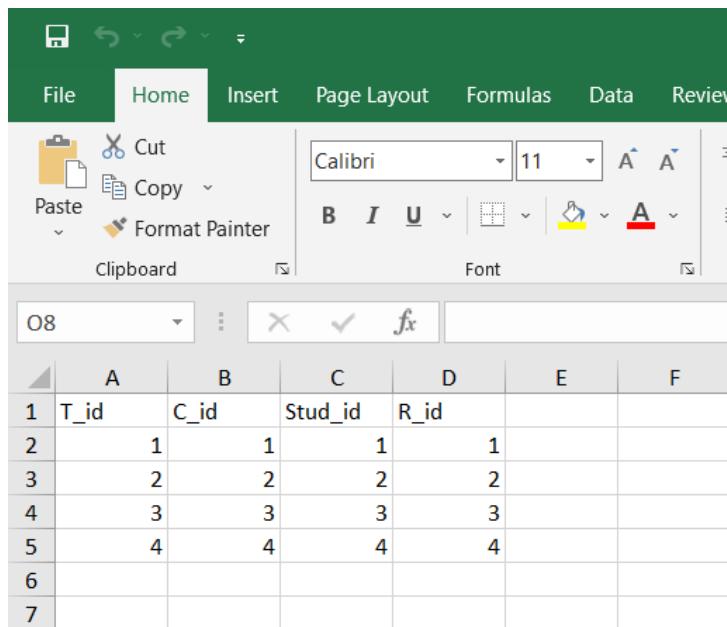
In data warehousing, a dimension is a collection of reference information about a measurable event. These events are known as facts and are stored in a fact table. Dimensions categorize and describe data warehouse facts and measures in ways that support meaningful answers to business questions. They form the very core of dimensional modeling.

Data Mart

A data mart is a simple form of data warehouse focused on a single subject or line of business. With a data mart, teams can access data and gain insights faster, because they don't have to spend time searching within a more complex data warehouse or manually aggregating data from different sources.

Steps:

- 1) Create fact and dimensions tables in excel(.xls) (Change sheet name for every sheet)

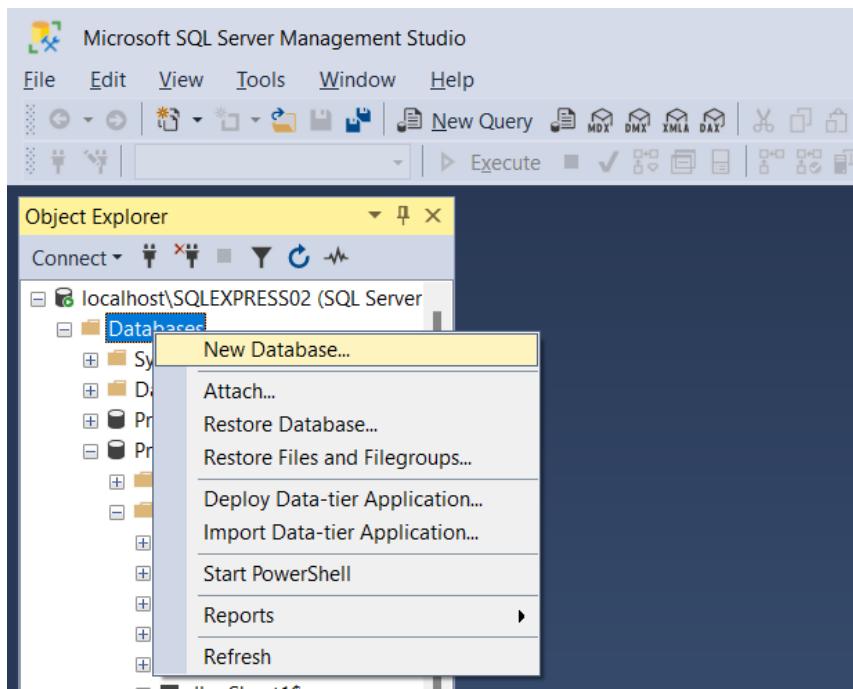


| | A | B | C | D | E | F |
|---|------|------|---------|------|---|---|
| 1 | T_id | C_id | Stud_id | R_id | | |
| 2 | | 1 | 1 | 1 | 1 | |
| 3 | | 2 | 2 | 2 | 2 | |
| 4 | | 3 | 3 | 3 | 3 | |
| 5 | | 4 | 4 | 4 | 4 | |
| 6 | | | | | | |
| 7 | | | | | | |

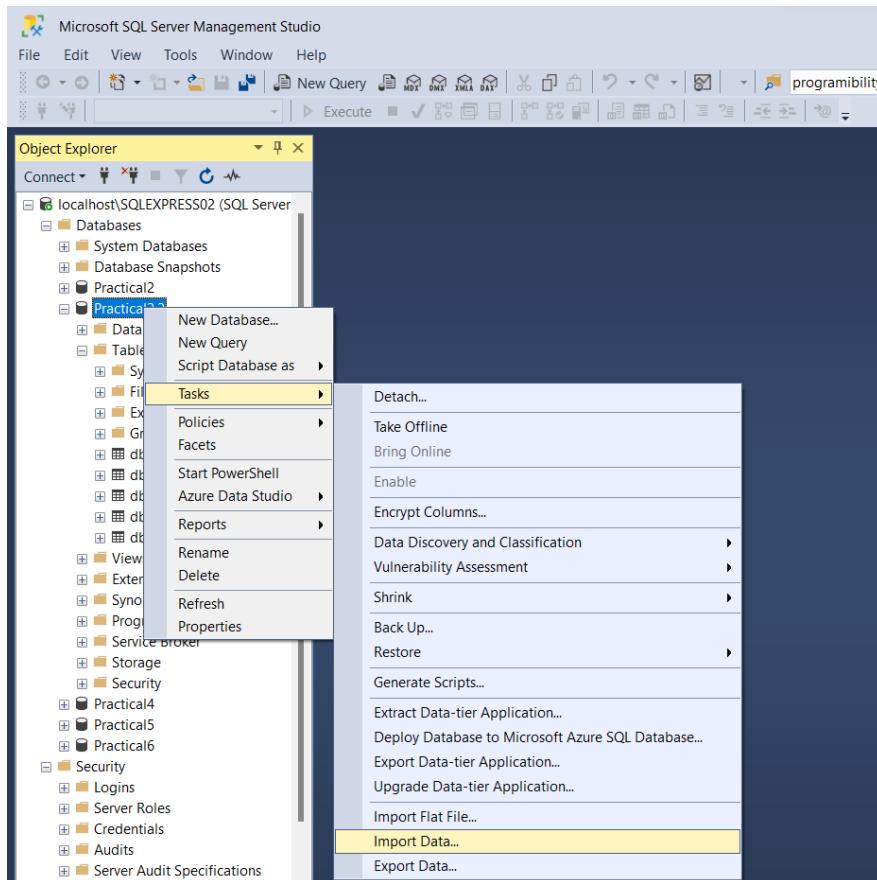
> This PC > Data (D:) > Practical-Msc-Sem-2 > BI > Excel-Tables

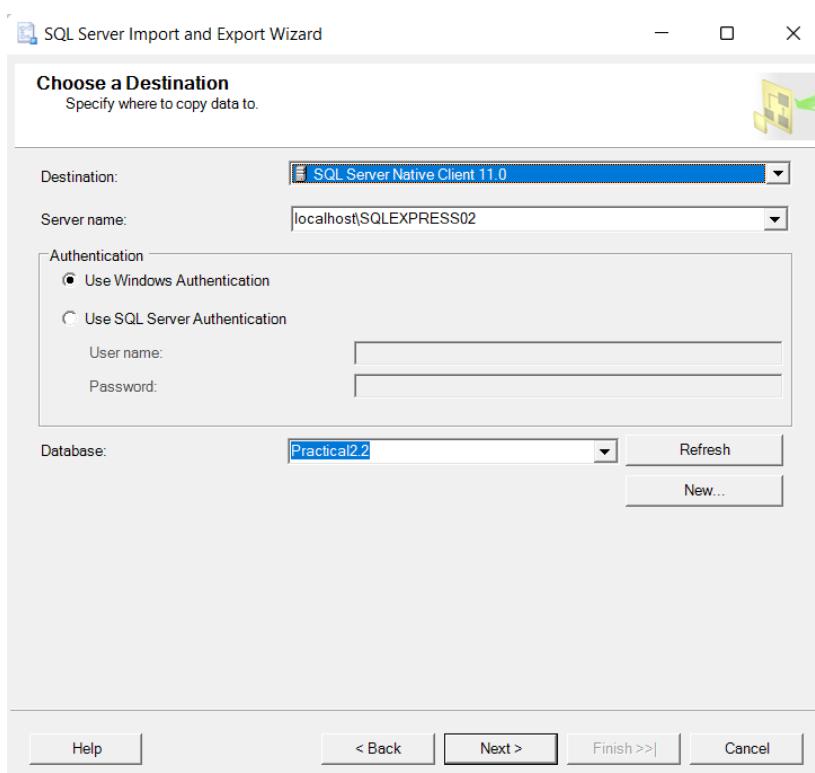
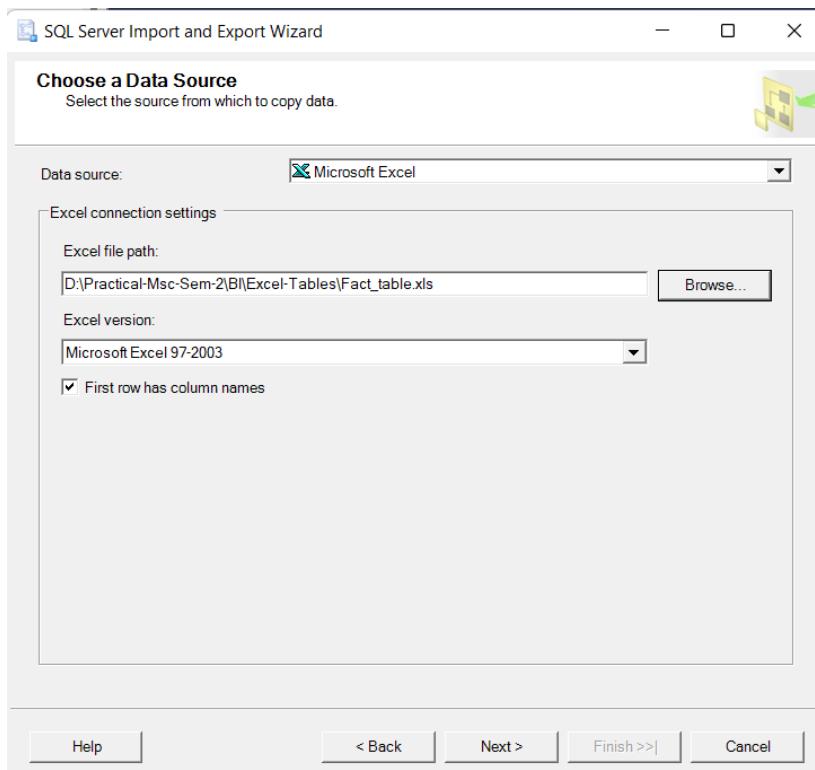
| Name | Date modified | Type | Size |
|-------------|------------------|-----------------------|-------|
| dimension_1 | 04-04-2022 08:09 | Microsoft Excel 97... | 25 KB |
| dimension_2 | 04-04-2022 08:20 | Microsoft Excel 97... | 25 KB |
| dimension_3 | 04-04-2022 08:20 | Microsoft Excel 97... | 25 KB |
| dimension_4 | 04-04-2022 08:20 | Microsoft Excel 97... | 25 KB |
| Fact_table | 04-04-2022 19:58 | Microsoft Excel 97... | 25 KB |

- 2) Create new database.

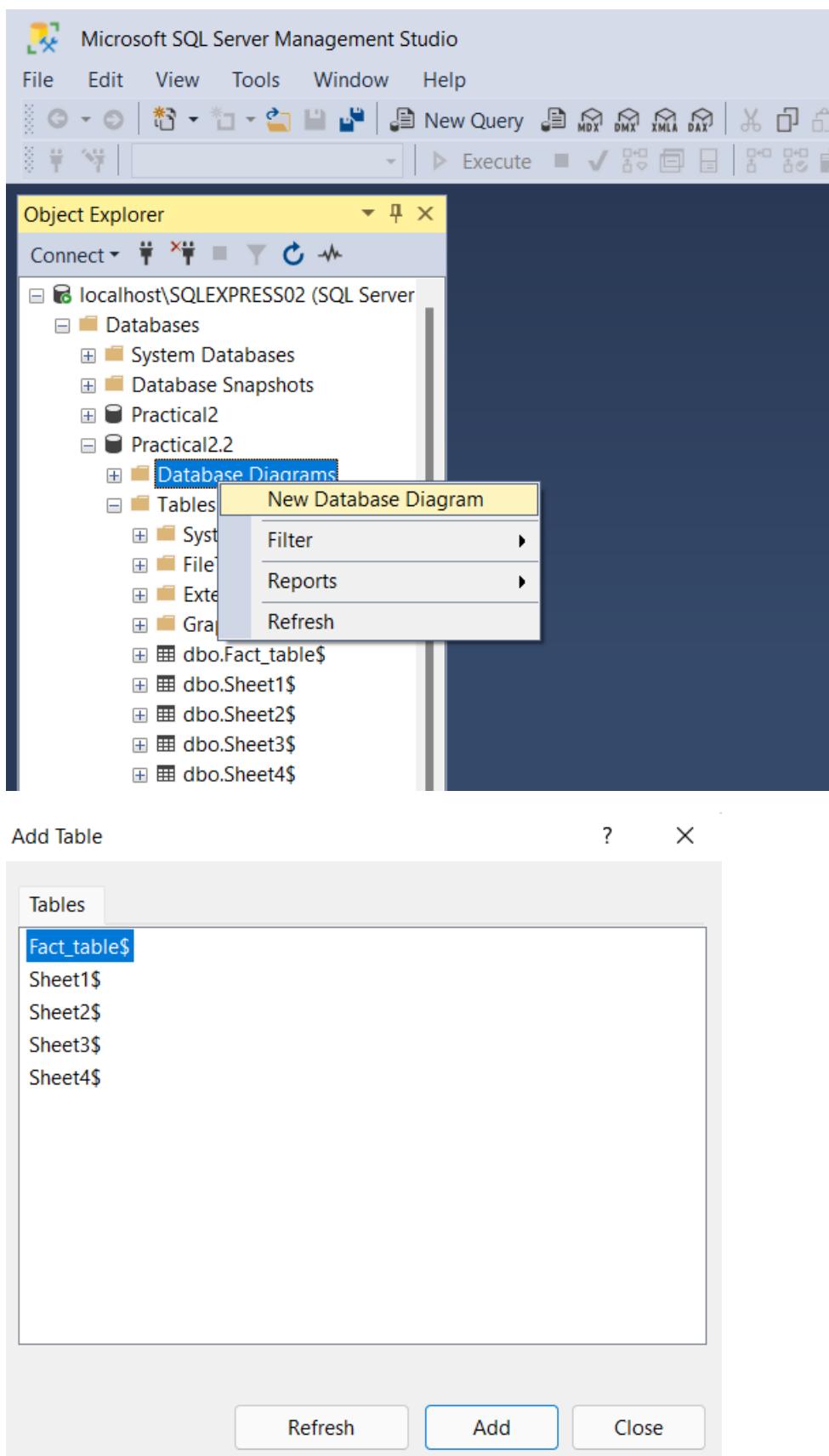


3) Import excel files > Choose data source > Choose destination source >Finish

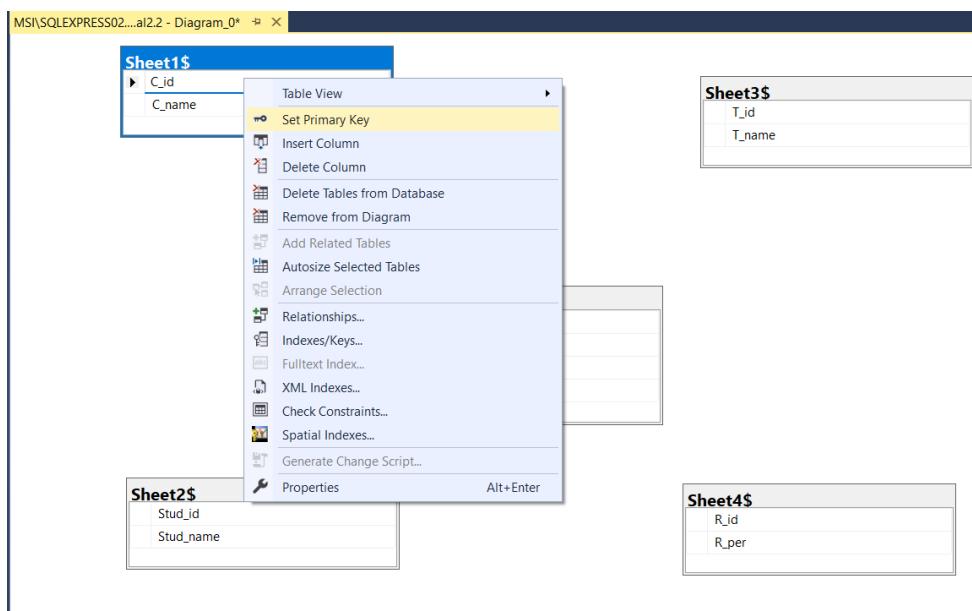




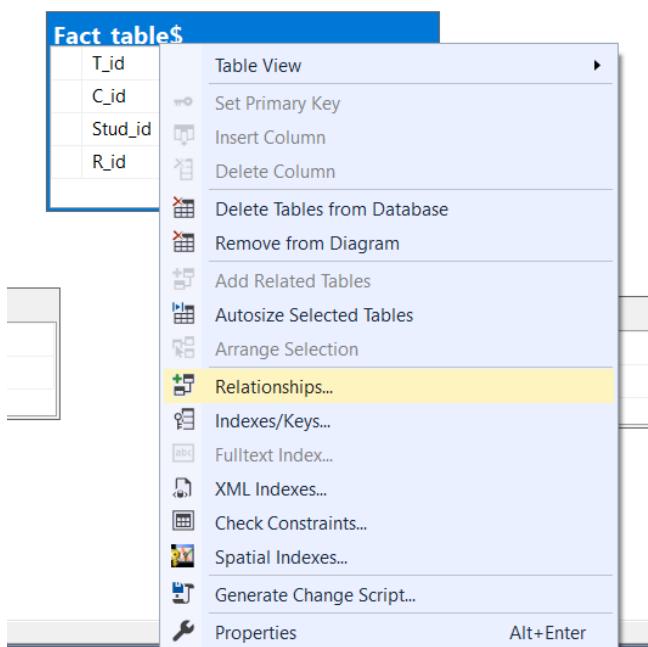
4) Create database diagrams

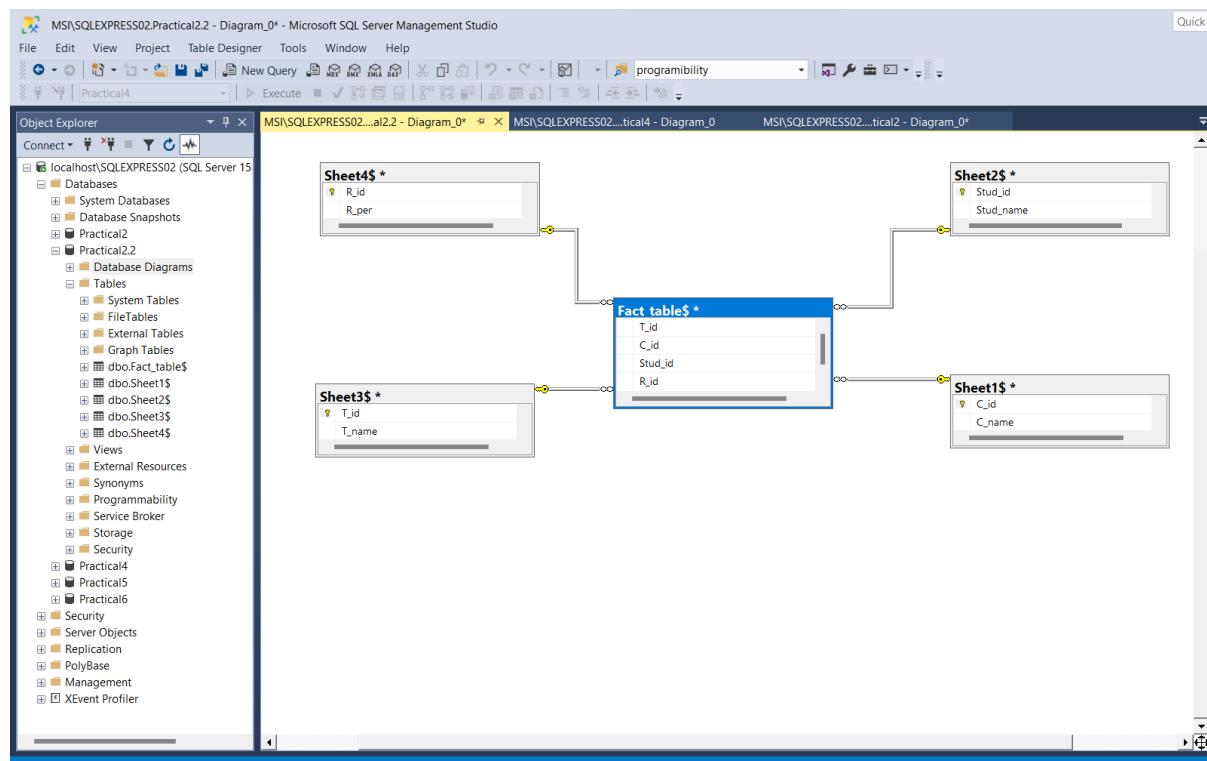
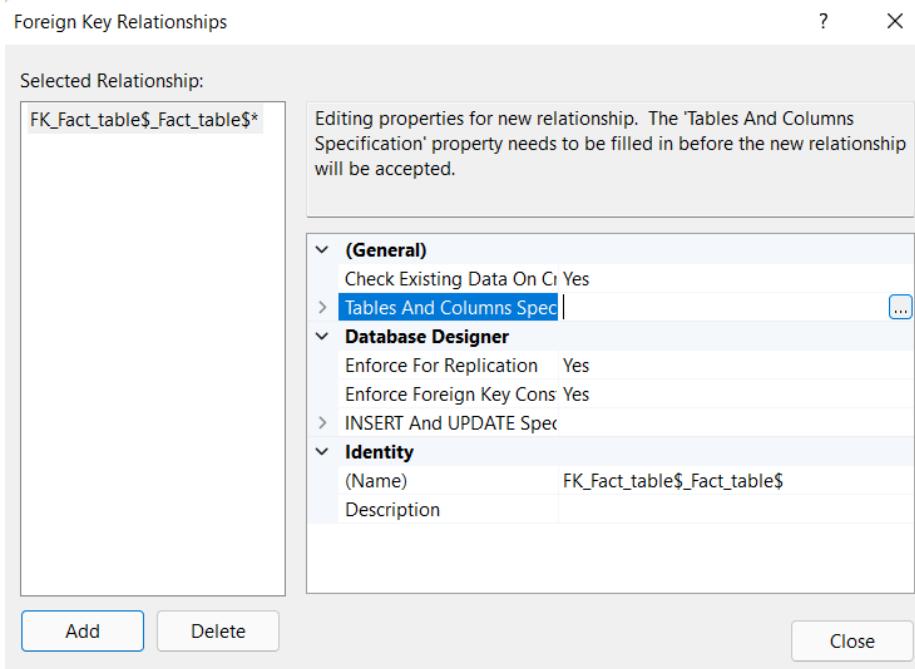


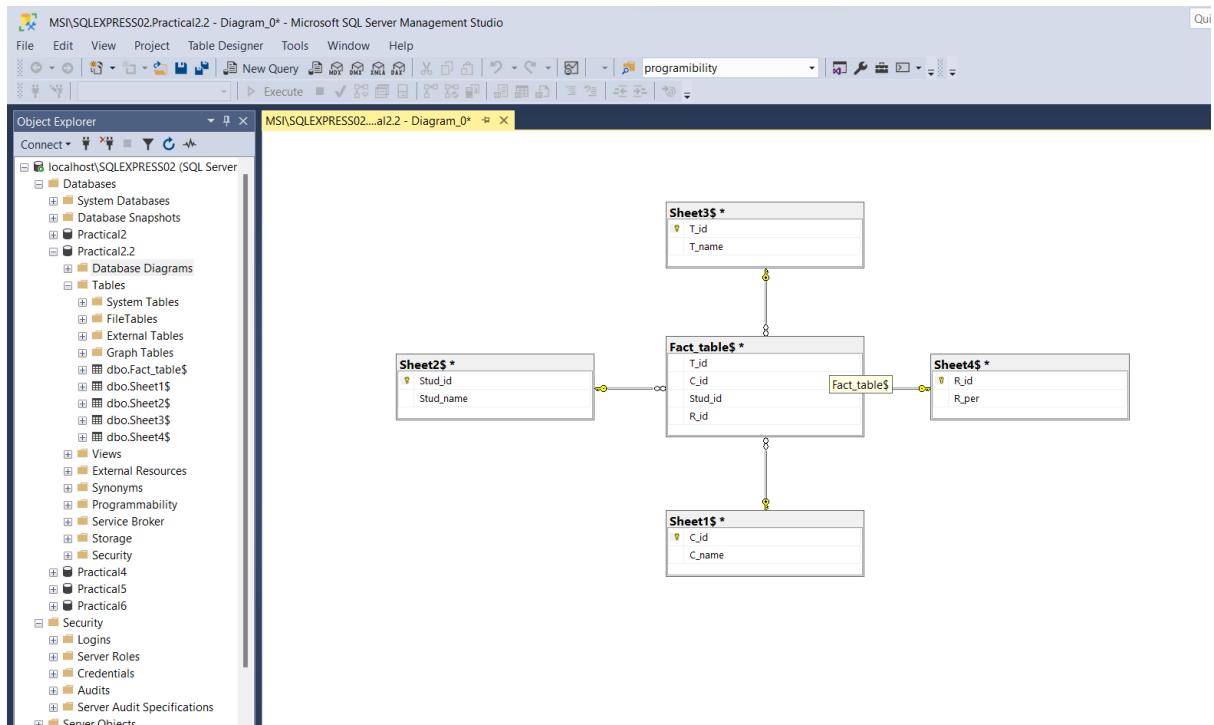
5) Set primary keys



6) Create relationships for fact table







Conclusion: Successfully Implemented fact, dimension table and data marts in MS SQL.

Practical 3

Aim: Develop an application to implement OLAP, roll-up, drill-down, slice, and dice operations.

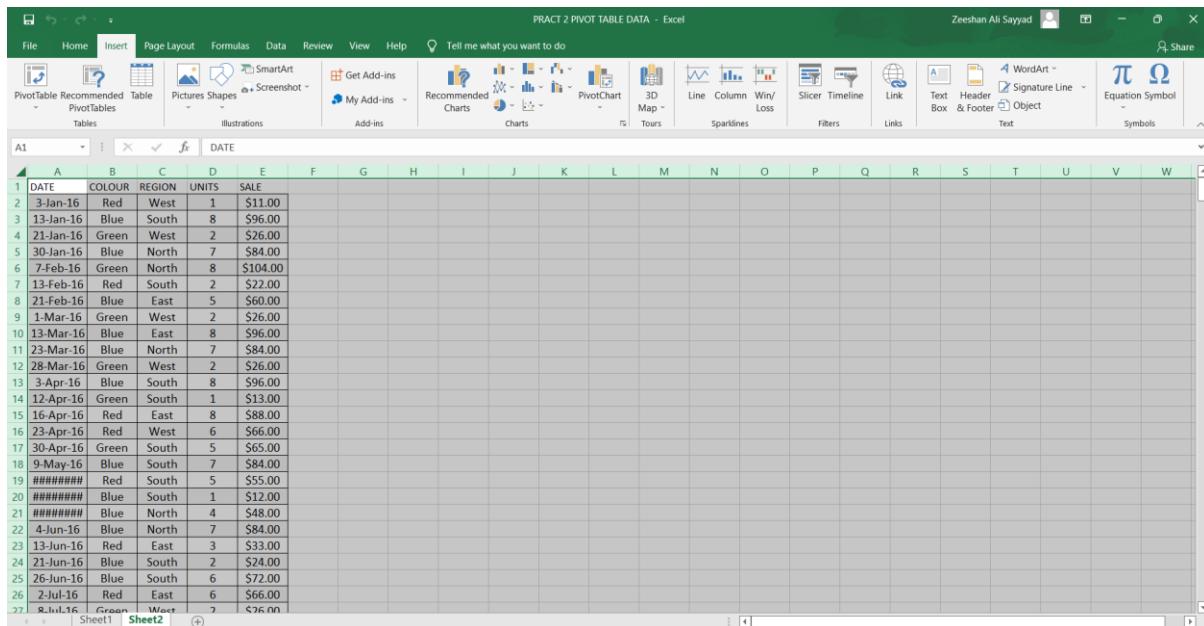
Theory:

A pivot table is a statistics tool that summarizes and reorganizes selected columns and rows of data in a spreadsheet or database table to obtain a desired report. The tool does not actually change the spreadsheet or database itself, it simply “pivots” or turns the data to view it from different perspectives.

Pivot tables are especially useful with large amounts of data that would be time-consuming to calculate by hand. A few data processing functions a pivot table can perform include identifying sums, averages, ranges or outliers. The table then arranges this information in a simple, meaningful layout that draws attention to key values.

Steps:

- 1) Select source data in excel sheet.



The screenshot shows a Microsoft Excel spreadsheet titled "PRACT 2 PIVOT TABLE DATA - Excel". The ribbon menu is visible at the top, with the "Insert" tab selected. The main area contains a data table with columns labeled DATE, COLOUR, REGION, UNITS, and SALE. The data spans from row 1 to 27, showing various entries for dates from January to July 2016, colors (Red, Blue, Green), regions (West, South, East, North), units sold, and sales amounts. Row 1 is a header row. The "Sheet1" tab is selected at the bottom.

| | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W |
|----|-----------|--------|--------|-------|----------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | DATE | COLOUR | REGION | UNITS | SALE | | | | | | | | | | | | | | | | | | |
| 2 | 3-Jan-16 | Red | West | 1 | \$11.00 | | | | | | | | | | | | | | | | | | |
| 3 | 13-Jan-16 | Blue | South | 8 | \$96.00 | | | | | | | | | | | | | | | | | | |
| 4 | 21-Jan-16 | Green | West | 2 | \$26.00 | | | | | | | | | | | | | | | | | | |
| 5 | 30-Jan-16 | Blue | North | 7 | \$84.00 | | | | | | | | | | | | | | | | | | |
| 6 | 7-Feb-16 | Green | North | 8 | \$104.00 | | | | | | | | | | | | | | | | | | |
| 7 | 13-Feb-16 | Red | South | 2 | \$22.00 | | | | | | | | | | | | | | | | | | |
| 8 | 21-Feb-16 | Blue | East | 5 | \$60.00 | | | | | | | | | | | | | | | | | | |
| 9 | 1-Mar-16 | Green | West | 2 | \$26.00 | | | | | | | | | | | | | | | | | | |
| 10 | 13-Mar-16 | Blue | East | 8 | \$96.00 | | | | | | | | | | | | | | | | | | |
| 11 | 23-Mar-16 | Blue | North | 7 | \$84.00 | | | | | | | | | | | | | | | | | | |
| 12 | 28-Mar-16 | Green | West | 2 | \$26.00 | | | | | | | | | | | | | | | | | | |
| 13 | 3-Apr-16 | Blue | South | 8 | \$96.00 | | | | | | | | | | | | | | | | | | |
| 14 | 12-Apr-16 | Green | South | 1 | \$13.00 | | | | | | | | | | | | | | | | | | |
| 15 | 16-Apr-16 | Red | East | 8 | \$88.00 | | | | | | | | | | | | | | | | | | |
| 16 | 23-Apr-16 | Red | West | 6 | \$66.00 | | | | | | | | | | | | | | | | | | |
| 17 | 30-Apr-16 | Green | South | 5 | \$65.00 | | | | | | | | | | | | | | | | | | |
| 18 | 9-May-16 | Blue | South | 7 | \$84.00 | | | | | | | | | | | | | | | | | | |
| 19 | ##### | Red | South | 5 | \$55.00 | | | | | | | | | | | | | | | | | | |
| 20 | ##### | Blue | South | 1 | \$12.00 | | | | | | | | | | | | | | | | | | |
| 21 | ##### | Blue | North | 4 | \$48.00 | | | | | | | | | | | | | | | | | | |
| 22 | 4-Jun-16 | Blue | North | 7 | \$84.00 | | | | | | | | | | | | | | | | | | |
| 23 | 13-Jun-16 | Red | East | 3 | \$33.00 | | | | | | | | | | | | | | | | | | |
| 24 | 21-Jun-16 | Blue | South | 2 | \$24.00 | | | | | | | | | | | | | | | | | | |
| 25 | 26-Jun-16 | Blue | South | 6 | \$72.00 | | | | | | | | | | | | | | | | | | |
| 26 | 2-Jul-16 | Red | East | 6 | \$66.00 | | | | | | | | | | | | | | | | | | |
| 27 | 8-Jul-16 | Green | West | 2 | \$26.00 | | | | | | | | | | | | | | | | | | |

- 2) Click on Insert tab > Pivot Table > Select source and location > Click Ok

PRACT 2 PIVOT TABLE DATA - Excel

PivotTable from table or range

Select a table or range
Table/Range: Sheet2!\$A:\$E

Choose where you want the PivotTable to be placed
New Worksheet (radio button not selected)
Existing Worksheet (radio button selected)

Location: Sheet2!\$K:\$K,Sheet2!\$L:\$L,Sheet2!\$M:\$M

Choose whether you want to analyze multiple tables
Add this data to the Data Model (checkbox not selected)

OK Cancel

3) Now Drop PivotTable Fields into Filters, Columns, Rows, Values Respectively.

PRACT 2 PIVOT TABLE DATA - Excel

PivotTable Tools

PivotTable Name: Active Field: PivotTable1 Options PivotTable

Active Field: Sum of UNITS

| A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S |
|----|-----------|--------|--------|-------|----------|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | DATE | COLOUR | REGION | UNITS | SALE | | | | | | | | | | | | | |
| 2 | 3-Jan-16 | Red | West | 1 | \$11.00 | | | | | | | | | | | | | |
| 3 | 13-Jan-16 | Blue | South | 8 | \$96.00 | | | | | | | | | | | | | |
| 4 | 21-Jan-16 | Green | West | 2 | \$26.00 | | | | | | | | | | | | | |
| 5 | 30-Jan-16 | Blue | North | 7 | \$84.00 | | | | | | | | | | | | | |
| 6 | 7-Feb-16 | Green | North | 8 | \$104.00 | | | | | | | | | | | | | |
| 7 | 13-Feb-16 | Red | South | 2 | \$22.00 | | | | | | | | | | | | | |
| 8 | 21-Feb-16 | Blue | East | 5 | \$60.00 | | | | | | | | | | | | | |
| 9 | 1-Mar-16 | Green | West | 2 | \$26.00 | | | | | | | | | | | | | |
| 10 | 13-Mar-16 | Blue | East | 8 | \$96.00 | | | | | | | | | | | | | |
| 11 | 23-Mar-16 | Blue | North | 7 | \$84.00 | | | | | | | | | | | | | |
| 12 | 28-Mar-16 | Green | West | 2 | \$26.00 | | | | | | | | | | | | | |
| 13 | 3-Apr-16 | Blue | South | 8 | \$96.00 | | | | | | | | | | | | | |
| 14 | 12-Apr-16 | Green | South | 1 | \$13.00 | | | | | | | | | | | | | |
| 15 | 16-Apr-16 | Red | East | 8 | \$88.00 | | | | | | | | | | | | | |
| 16 | 23-Apr-16 | Red | West | 6 | \$66.00 | | | | | | | | | | | | | |
| 17 | 30-Apr-16 | Green | South | 5 | \$65.00 | | | | | | | | | | | | | |
| 18 | 9-May-16 | Blue | South | 7 | \$84.00 | | | | | | | | | | | | | |
| 19 | ##### | Red | South | 5 | \$55.00 | | | | | | | | | | | | | |
| 20 | ##### | Blue | South | 1 | \$12.00 | | | | | | | | | | | | | |
| 21 | ##### | Blue | North | 4 | \$48.00 | | | | | | | | | | | | | |
| 22 | 4-Jun-16 | Blue | North | 7 | \$84.00 | | | | | | | | | | | | | |
| 23 | 13-Jun-16 | Red | East | 3 | \$33.00 | | | | | | | | | | | | | |
| 24 | 21-Jun-16 | Blue | South | 2 | \$24.00 | | | | | | | | | | | | | |
| 25 | 26-Jun-16 | Blue | South | 6 | \$72.00 | | | | | | | | | | | | | |
| 26 | 2-Jul-16 | Red | East | 6 | \$66.00 | | | | | | | | | | | | | |
| 27 | 8-Jul-16 | Green | West | 2 | \$26.00 | | | | | | | | | | | | | |

PivotTable Fields

Choose fields to add to report:
Search

- DATE
- COLOUR
- REGION
- UNITS
- SALE
- Quarters
- Years

More Tables...

Drag fields between areas below:

Filters
COLOUR
DATE

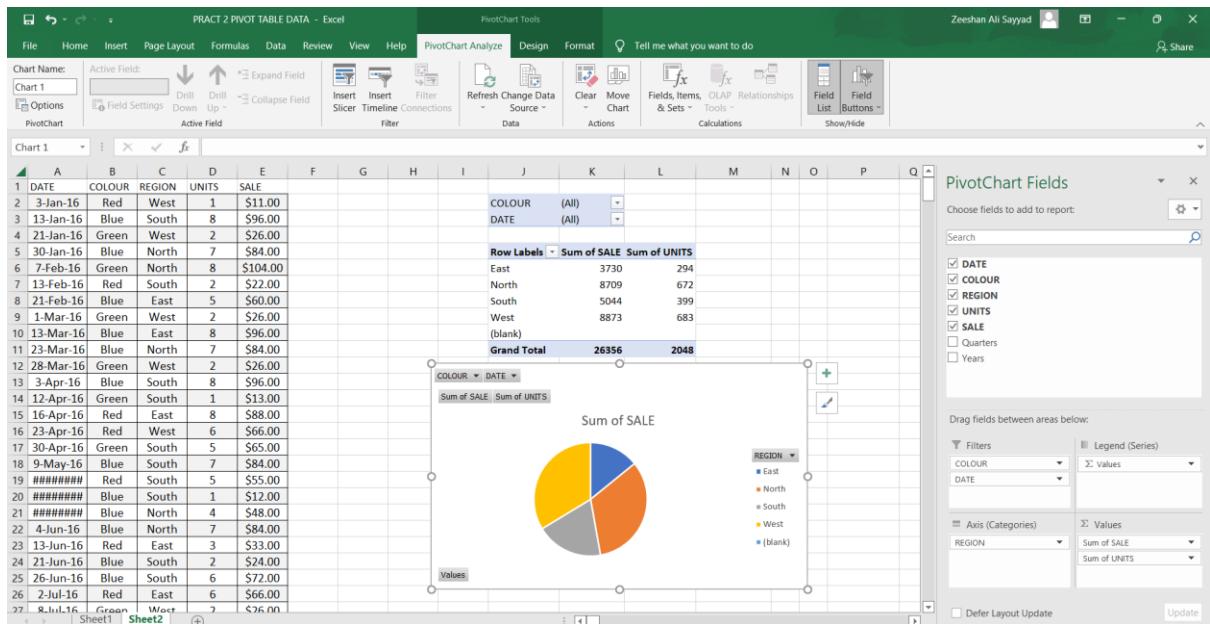
Columns
Sum of SALE
Sum of UNITS

Rows
REGION
Sum of SALE
Sum of UNITS

Values
Sum of UNITS

Defer Layout Update Update

4) Insert Pie Chart or Bar graph.



Conclusion: Pivot table implemented successfully.

Practical 4

Aim: Develop an application to construct a multidimensional data.

Theory:

What is Multi-Dimensional Data Model?

A multidimensional model views data in the form of a data-cube. A data cube enables data to be modeled and viewed in multiple dimensions. It is defined by dimensions and facts.

The dimensions are the perspectives or entities concerning which an organization keeps records. For example, a shop may create a sales data warehouse to keep records of the store's sales for the dimension time, item, and location. These dimensions allow the user to keep track of things, for example, monthly sales of items and the locations at which the items were sold. Each dimension has a table related to it, called a dimensional table, which describes the dimension further. For example, a dimensional table for an item may contain the attributes item_name, brand, and type.

A multidimensional data model is organized around a central theme, for example, sales. This theme is represented by a fact table. Facts are numerical measures. The fact table contains the names of the facts or measures of the related dimensional tables.

#Steps:

1) Create new database.

```
CREATE TABLE DIM_customer
```

```
(  
    custid VARCHAR(6),  
    fname VARCHAR(30),  
    mname VARCHAR(30),  
    lname VARCHAR(30),  
    city VARCHAR(15),  
    mobileno VARCHAR(10),  
    occupation VARCHAR(10),  
    dob DATE
```

```
)
```

```
CREATE TABLE DIM_branch
```

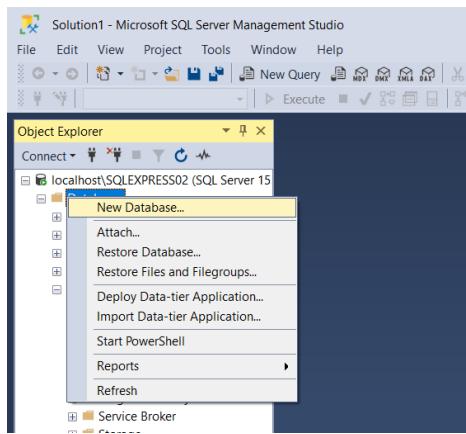
```
(  
    bid VARCHAR(6),  
    bname VARCHAR(30),
```

```
        bcity VARCHAR(30),  
    )  
  
CREATE TABLE DIM_account  
(  
    acnumber VARCHAR(6),  
    custid  VARCHAR(6),  
    bid VARCHAR(6),  
    opening_balance VARCHAR(7),  
    aod DATE,  
    atype VARCHAR(10),  
    astatus VARCHAR(10)  
)  
  
CREATE TABLE DIM_trandetails  
(  
    tnumber VARCHAR(6),  
    acnumber VARCHAR(6),  
    dot DATE,  
    medium_of_transaction VARCHAR(20),  
    transaction_type VARCHAR(20),  
    transaction_amount VARCHAR(7)  
)  
  
CREATE TABLE DIM_loan  
(  
    loan_id VARCHAR(10),  
    custid VARCHAR(6),  
    bid VARCHAR(6),  
    loan_amount VARCHAR(7)  
)  
  
CREATE TABLE FACT_BANK  
(
```

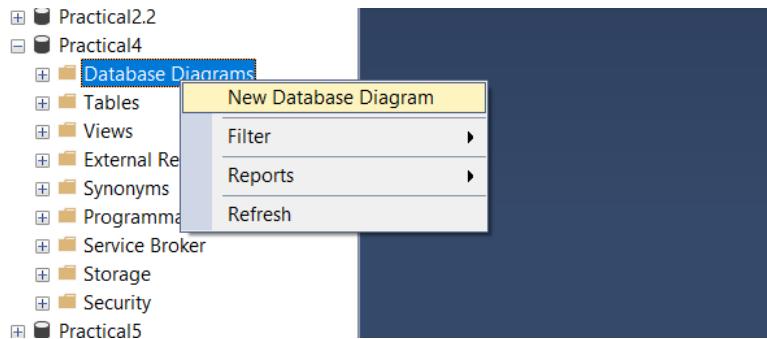
```

        custid VARCHAR(6),
        bid VARCHAR(6),
        acnumber VARCHAR(6),
        tnumber VARCHAR(6),
        loan_id VARCHAR(10)
    )

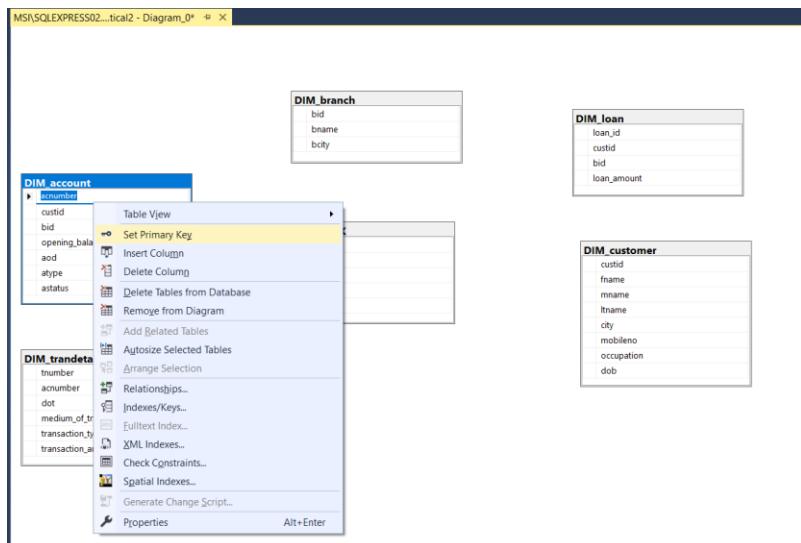
```



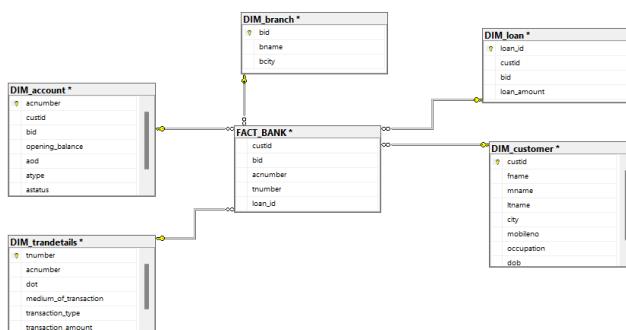
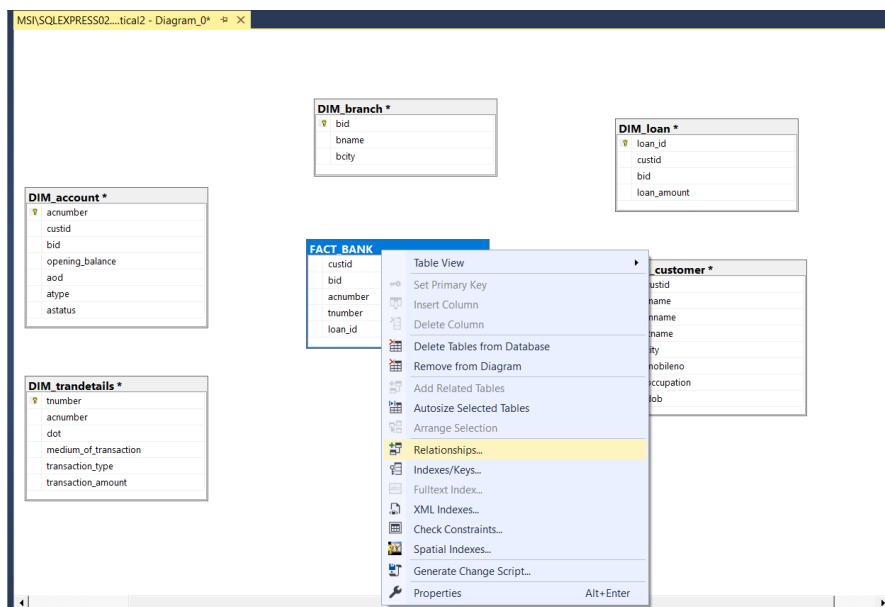
2) Create database diagram.



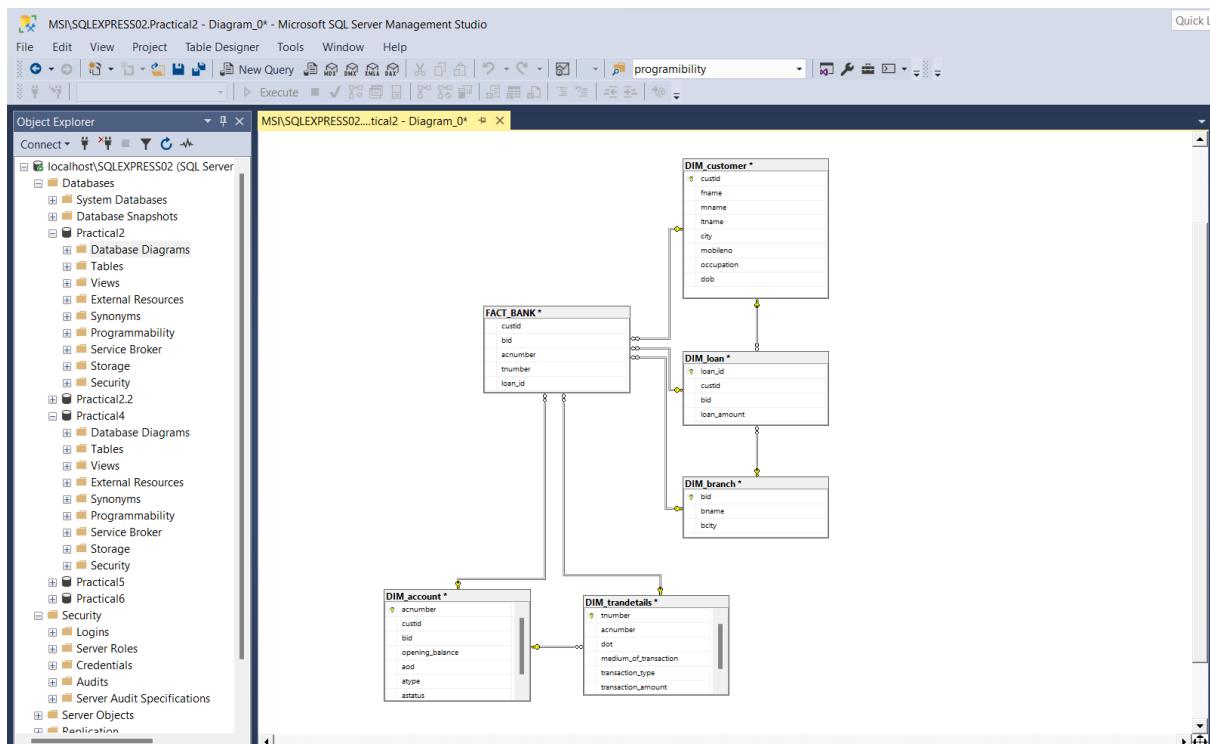
3) Set primary key for each table:



4) Set Relationships for fact table.



5) Set relationships of dimensional table with each other.



Conclusion: Multidimensional data constructed successfully.

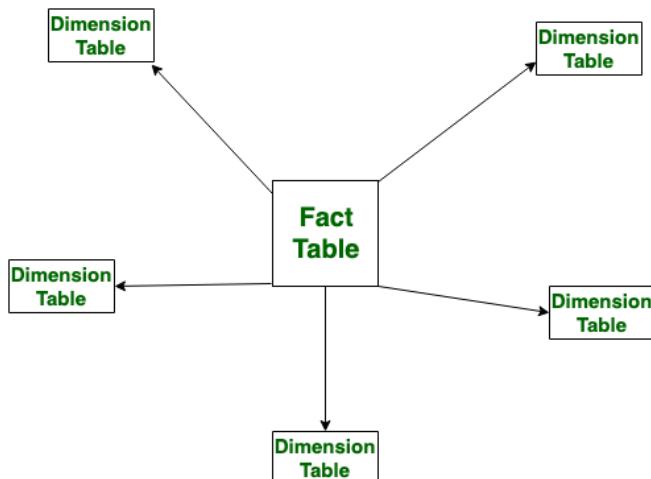
Practical 5

Aim: Design and create cube by identifying measures and dimensions for star schema.

Theory:

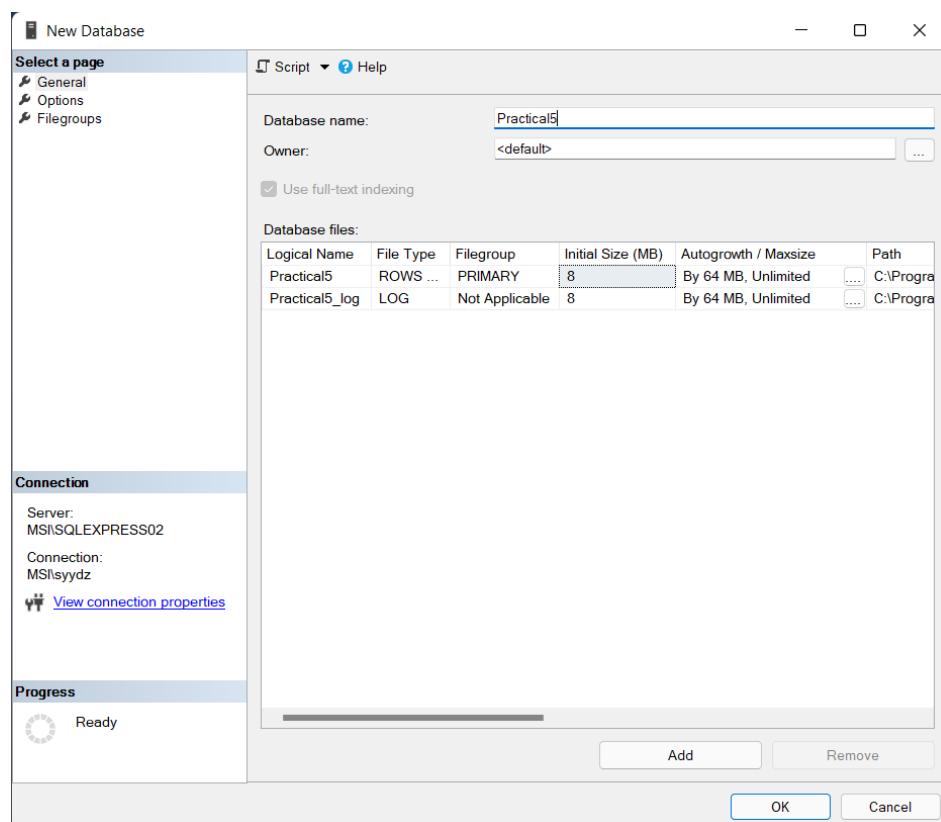
Star Schema:

Star schema is the type of multidimensional model which is used for data warehouse. In star schema, the fact tables and the dimension tables are contained. In this schema fewer foreign-key join is used. This schema forms a star with fact table and dimension tables.

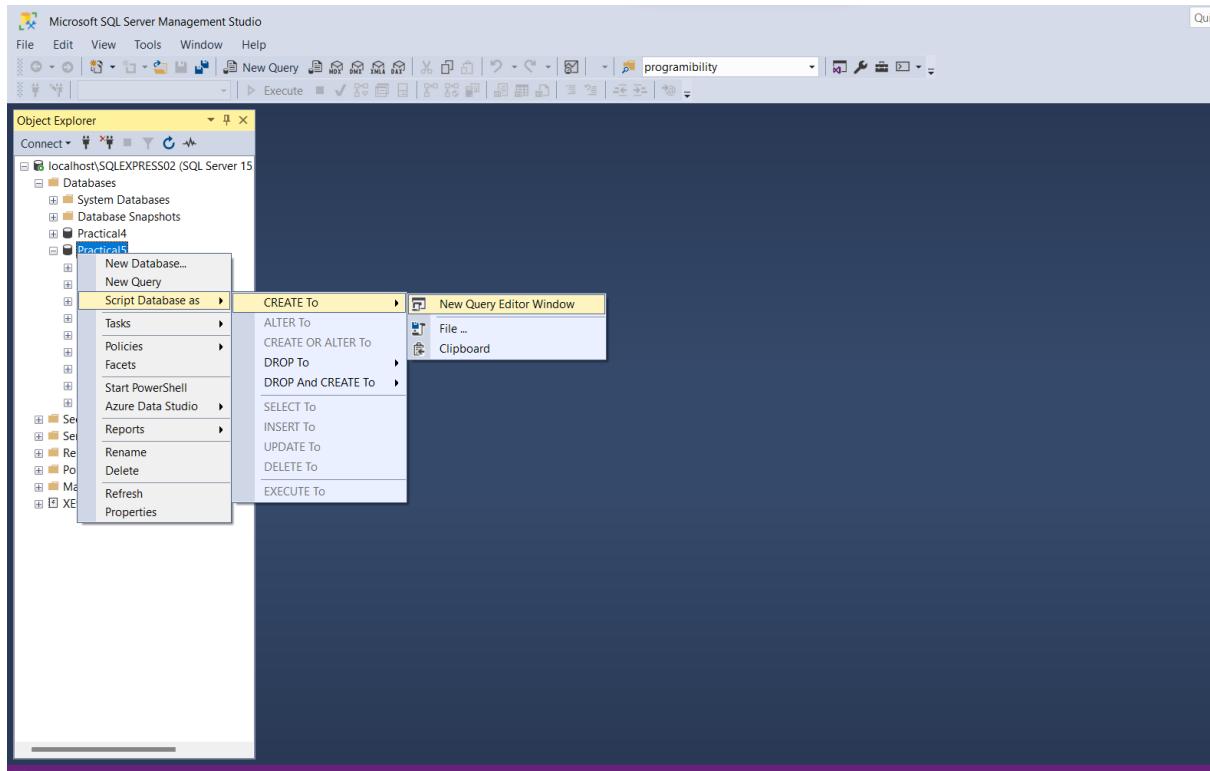


#Steps

1) Create New Database



2) Right Click on Database > Script Database as > Create to > Select: New Query Editor Window.



3) Remove pre-written queries

4) Create tables:

```
CREATE TABLE DIM_EMPLOYEE
(EMP_ID INT,
EMP_NAME VARCHAR(25))

CREATE TABLE DIM_BRANCH
(BRANCH_ID INT,
BRANCH_NAME VARCHAR(25))

CREATE TABLE DIM_PRODUCT
(PROD_ID INT,
PROD_NAME VARCHAR(25))

CREATE TABLE DIM_CUSTOMER
(CUST_ID INT,
CUST_NAME VARCHAR(25))

CREATE TABLE FACT_SHOP
(EMP_ID INT,
BRANCH_ID INT,
PROD_ID INT,
CUST_ID INT)
```

4) Execute queries

The screenshot shows the Microsoft SQL Server Management Studio interface. The title bar reads "SQLQuery1.sql - localhost\SQLEXPRESS02.Practical5 (MSI\sydz (54)) - Microsoft SQL Server Management Studio". The main window displays a script titled "Execute (F5)" containing several CREATE TABLE statements for dimensions and facts. The "Messages" pane at the bottom indicates that the commands completed successfully with a completion time of 2022-04-03T22:55:45.9780615+05:30. A status bar at the bottom right shows "localhost\SQLEXPRESS02 (15.... MSI\sydz (54) Practical5 00:00:00 | 0 rows".

```
CREATE TABLE DIM_EMPLOYEE
(EMP_ID INT,
EMP_NAME VARCHAR(25))

CREATE TABLE DIM_BRANCH
(BRANCH_ID INT,
BRANCH_NAME VARCHAR(25))

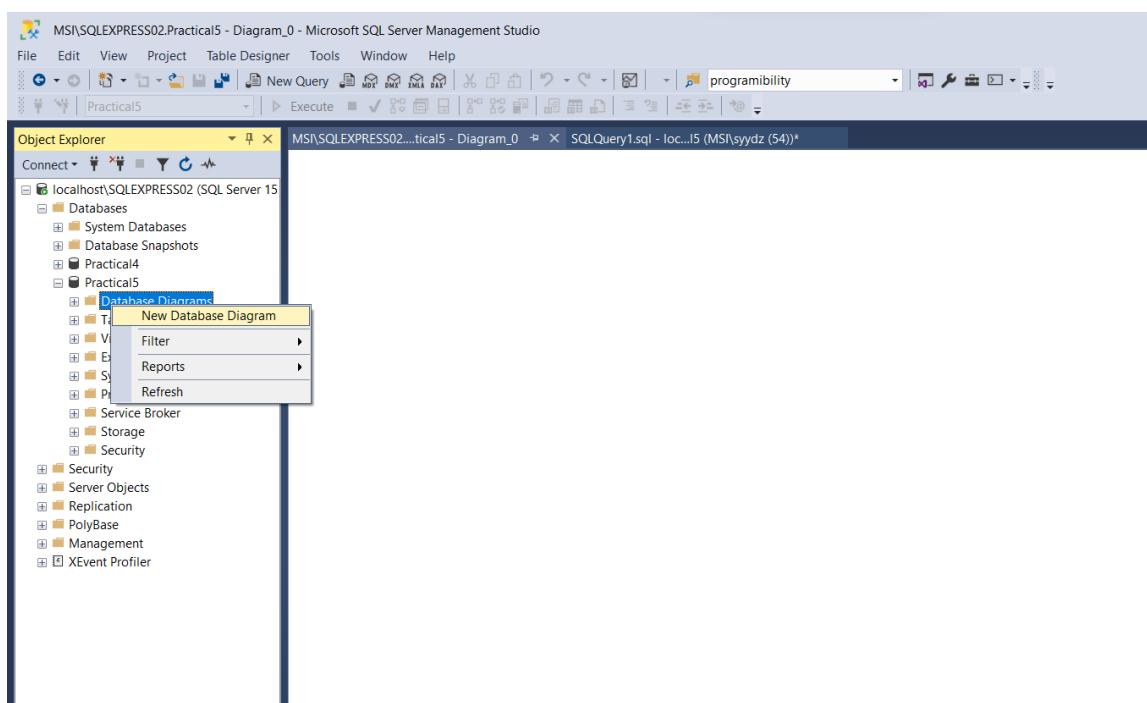
CREATE TABLE DIM_PRODUCT
(PROD_ID INT,
PROD_NAME VARCHAR(25))

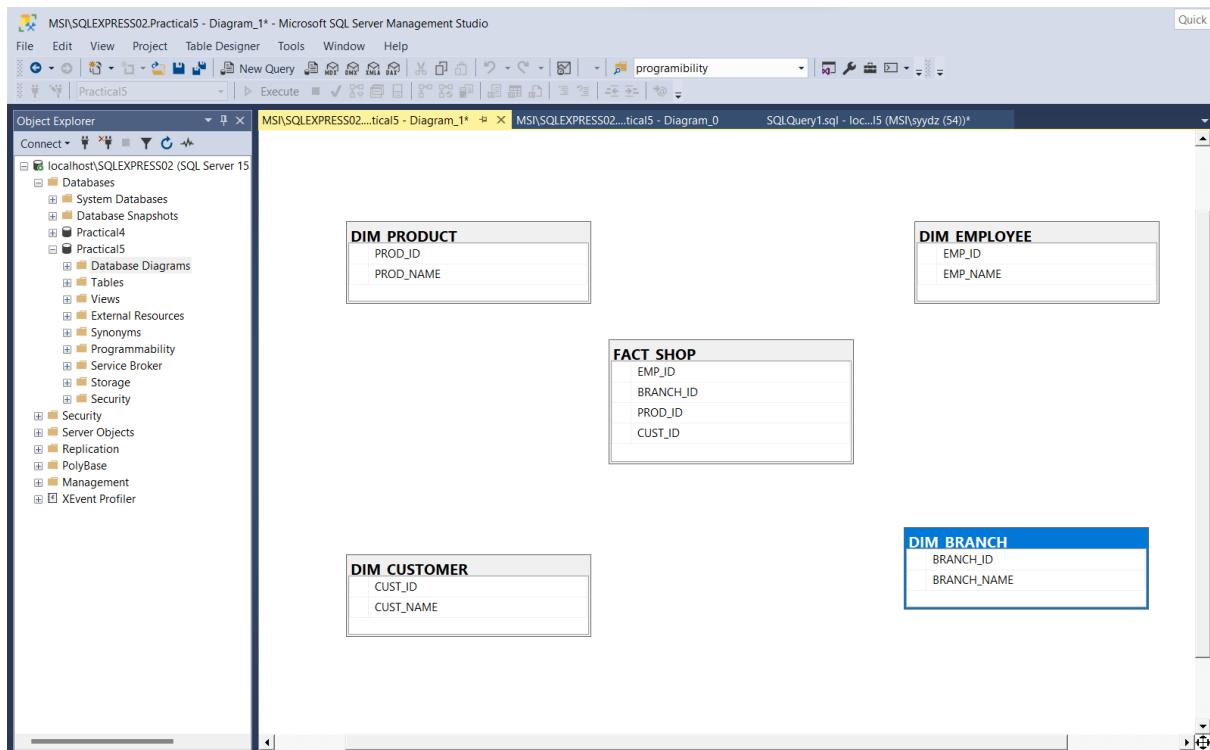
CREATE TABLE DIM_CUSTOMER
(CUST_ID INT,
CUST_NAME VARCHAR(25))

CREATE TABLE FACT_SHOP
(EMP_ID INT,
BRANCH_ID INT,
PROD_ID INT,
CUST_ID INT)
```

6) Create Database diagrams:

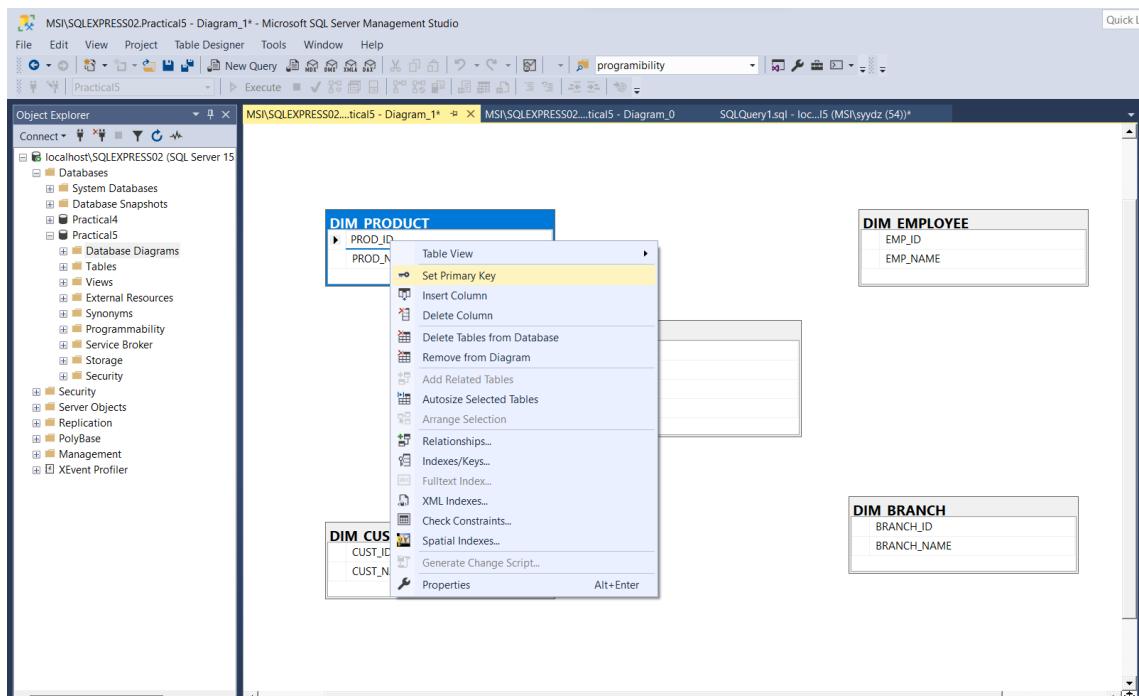
Right Click On Database Diagrams > New Database Diagram





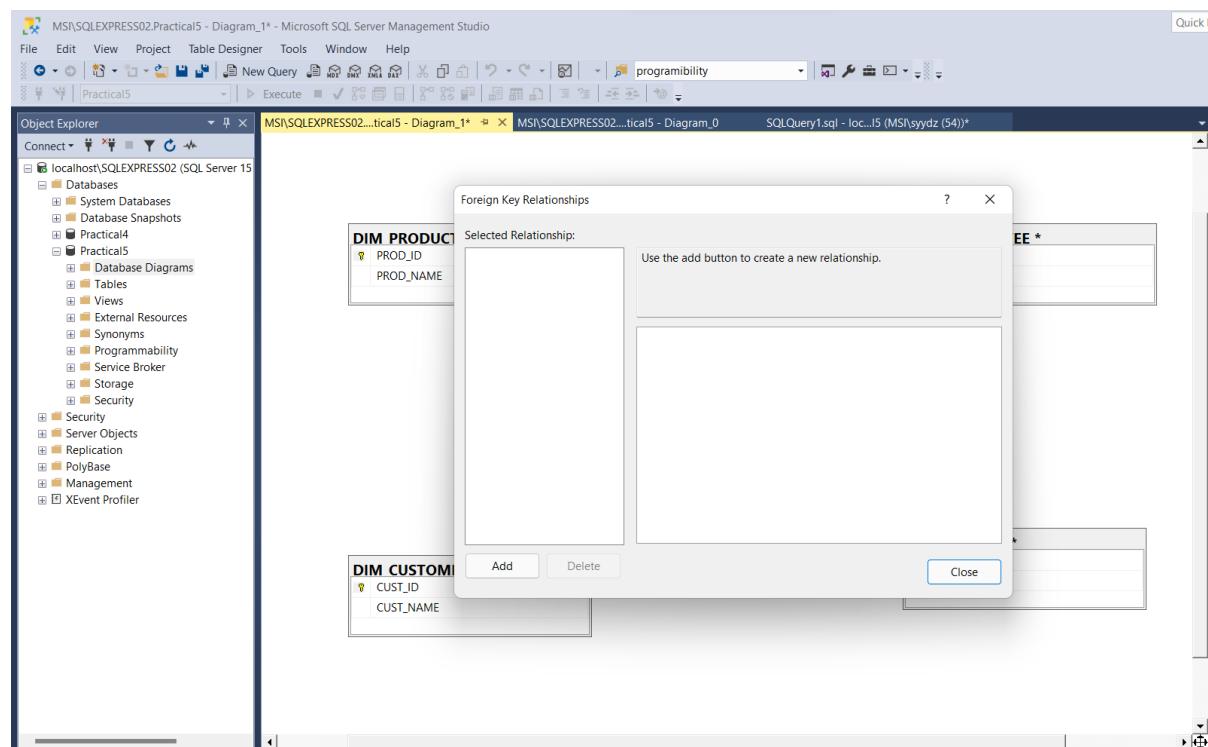
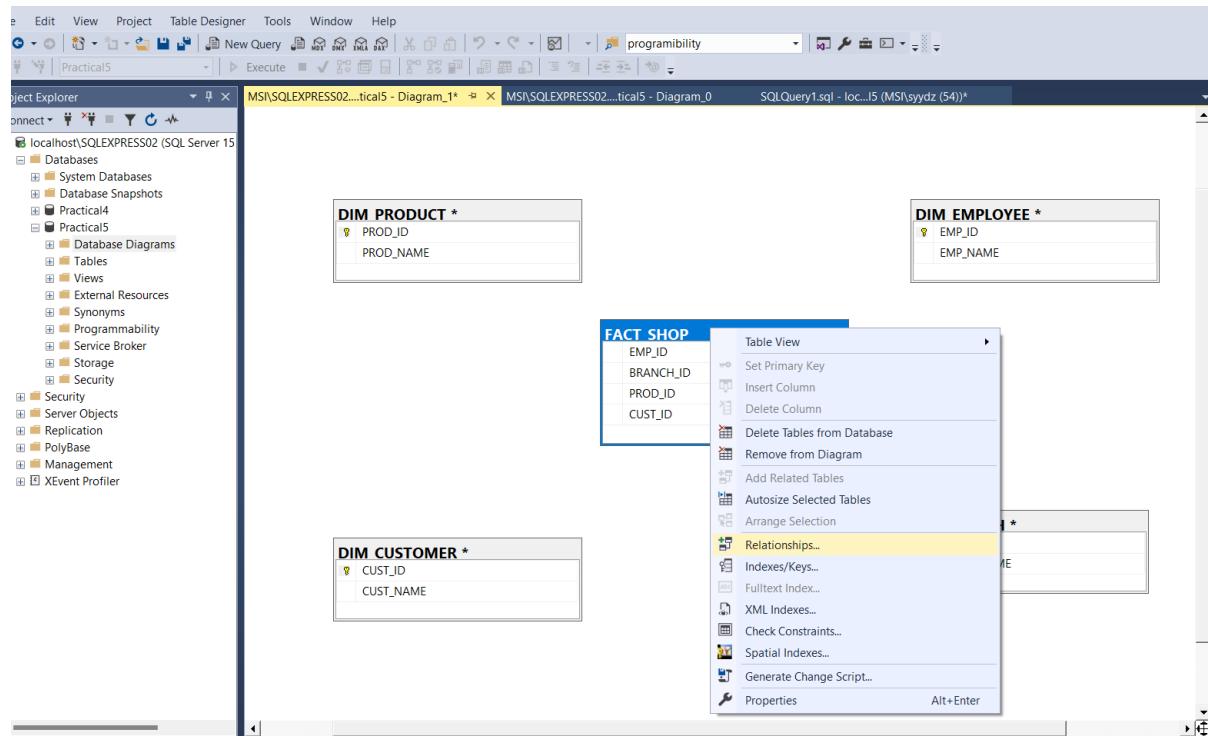
7) Set primary keys :

PROD_ID, EMP_ID, CUST_ID, BRANCH_ID

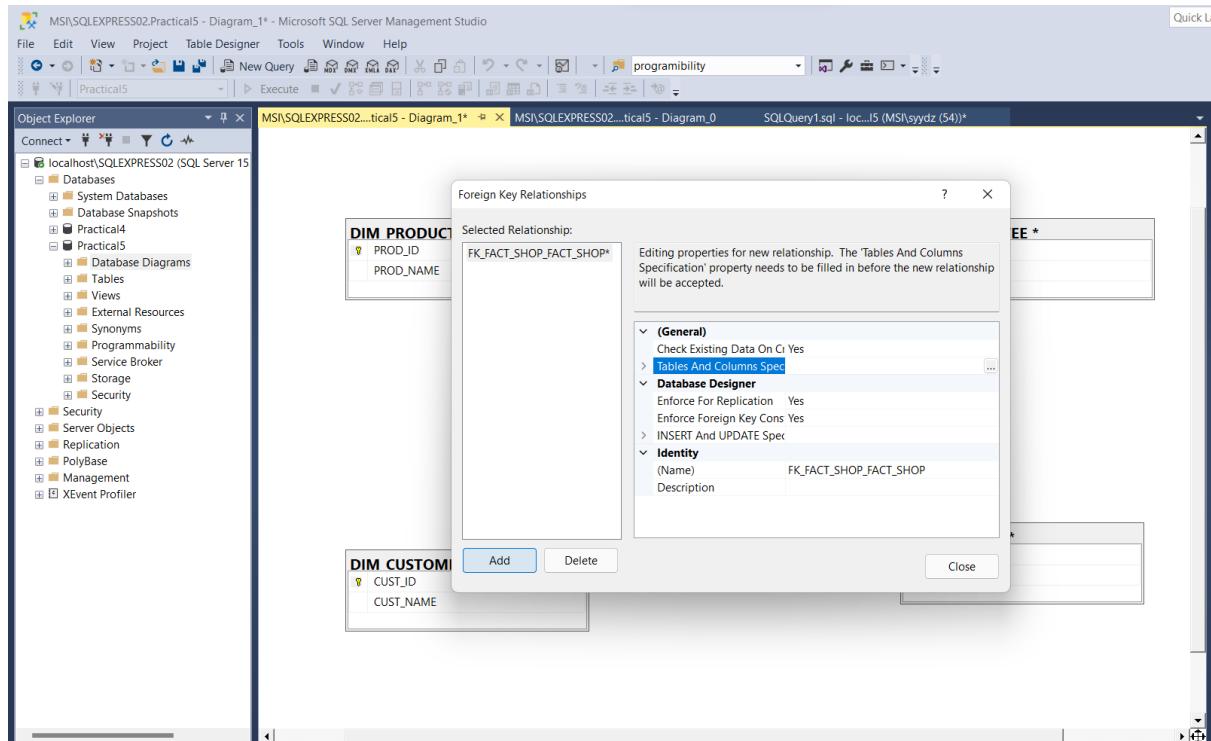


8) Create Relationships:

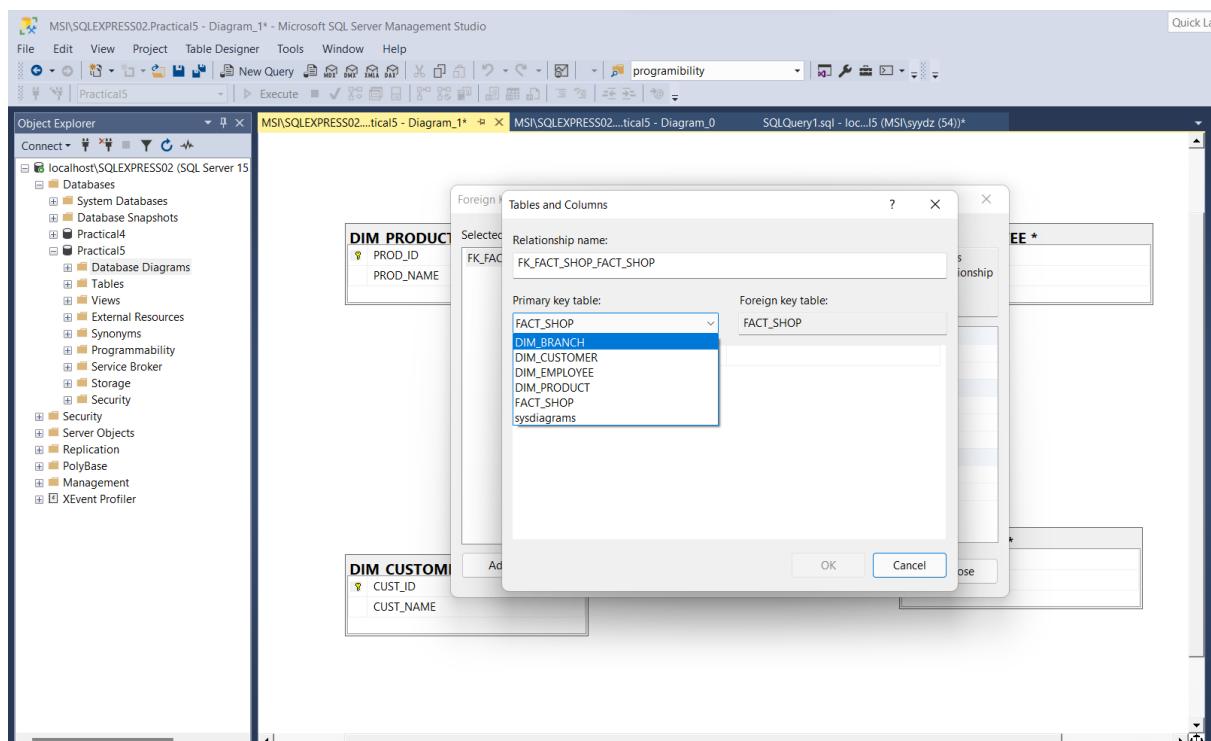
Right Click on Fact Shop table > Select Relationships > Add relations.

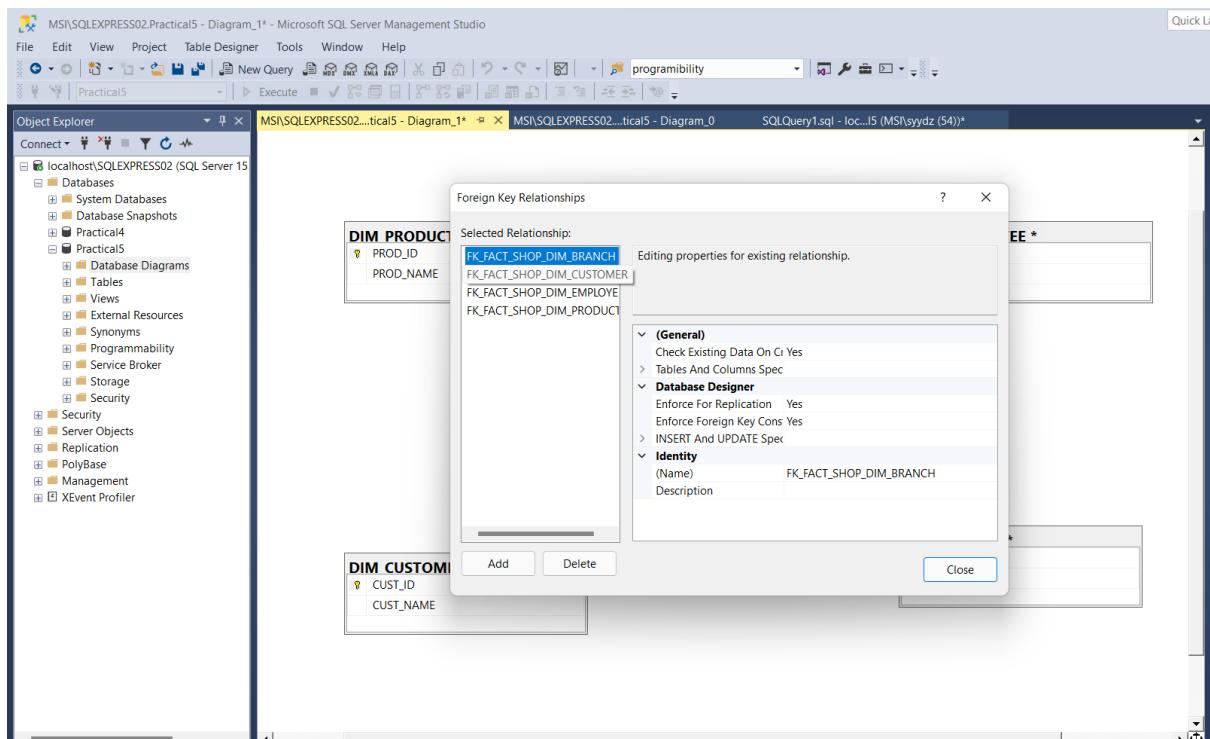
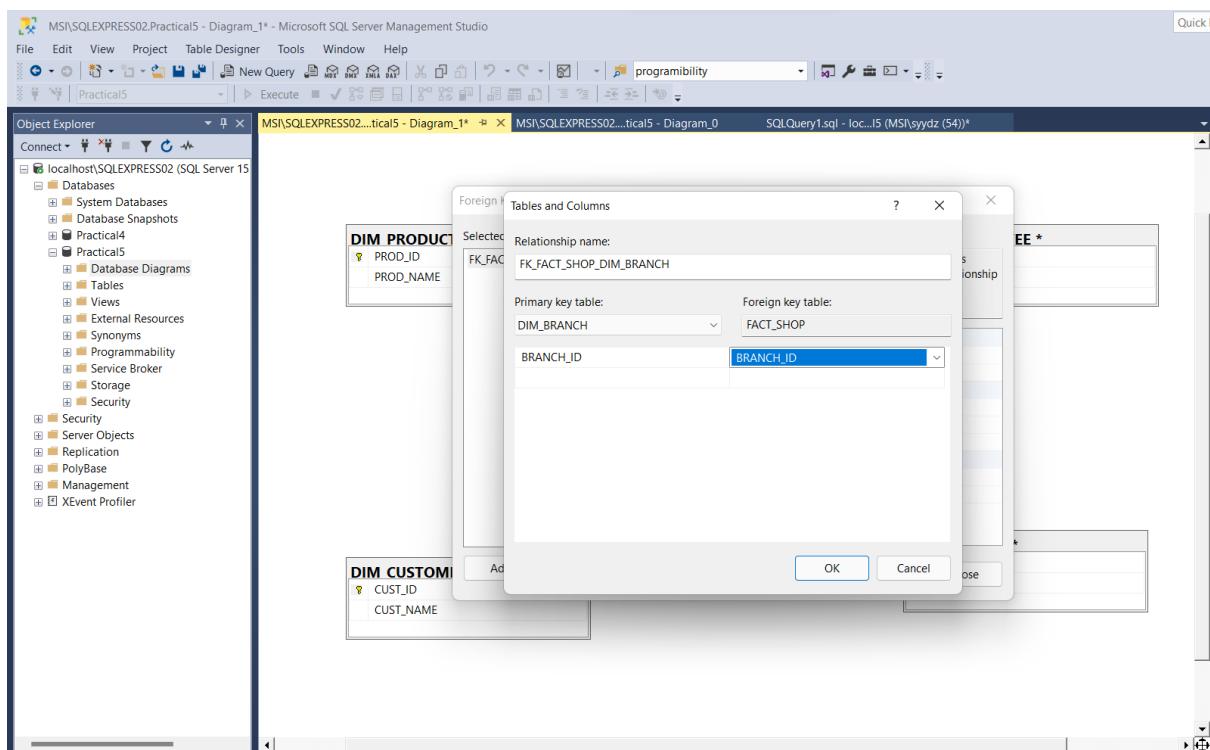


8) Click on Tables And Columns Sec(Expand ...)

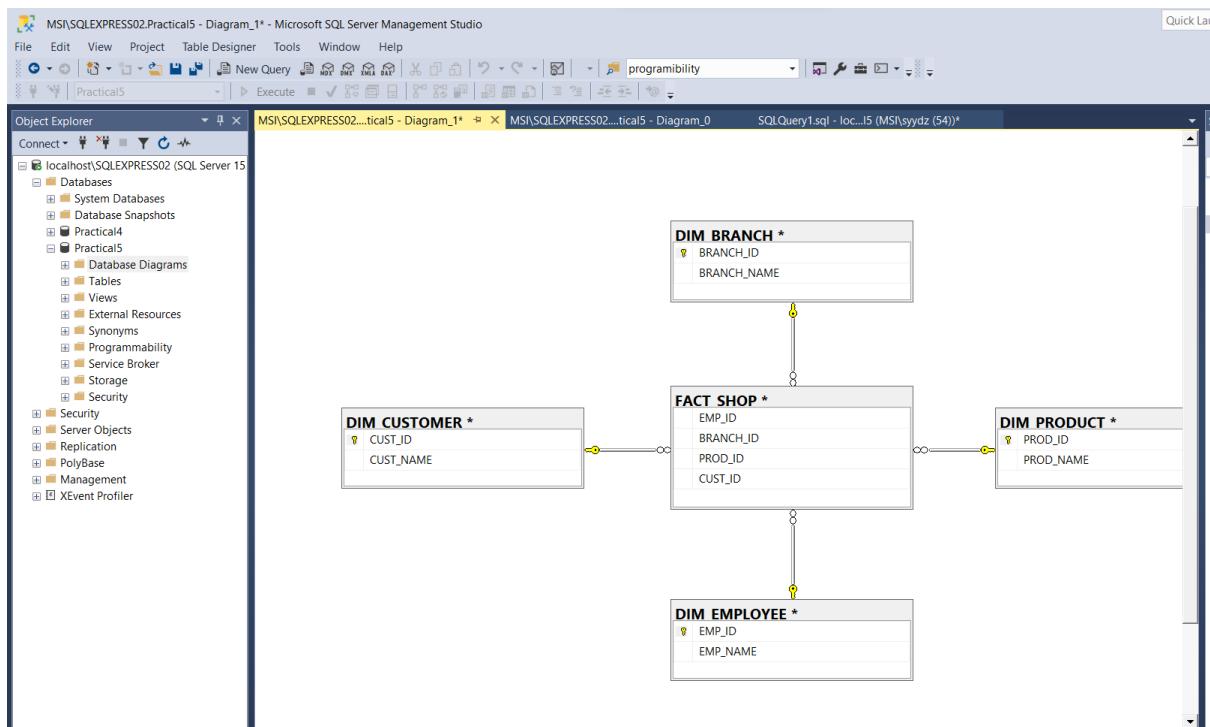
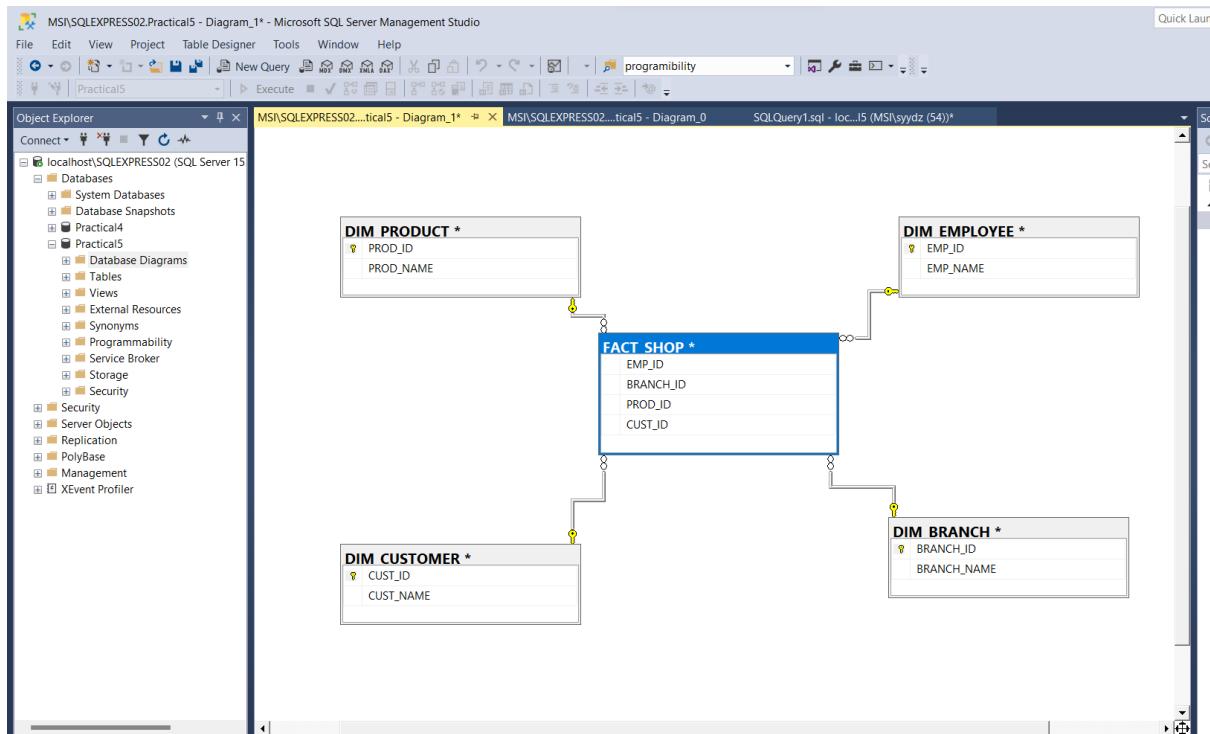


9) Select Primary key tables for every relation





10) Close Relationships window



Conclusion: Star schema executed successfully.

Practical 6

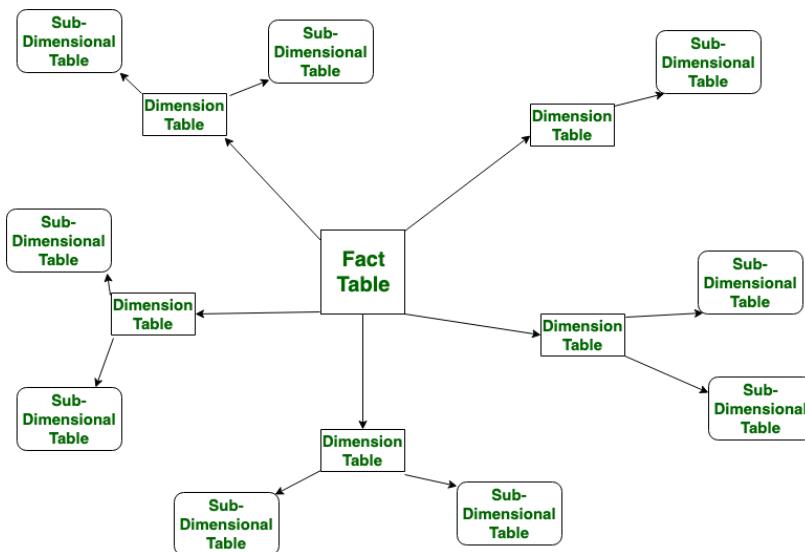
Aim: Design and create cube by identifying measures and dimensions for snowflake

schema.

Theory:

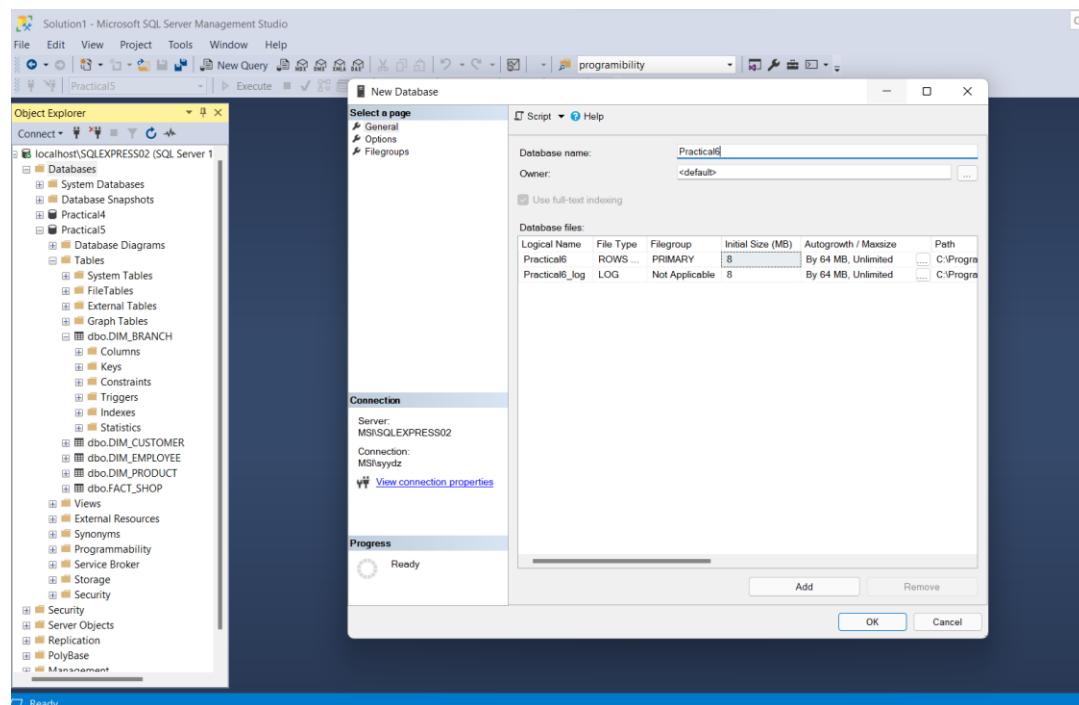
Snowflake Schema:

Snowflake Schema is also the type of multidimensional model which is used for data warehouse. In snowflake schema, The fact tables, dimension tables as well as sub dimension tables are contained. This schema forms a snowflake with fact tables, dimension tables as well as sub-dimension tables.

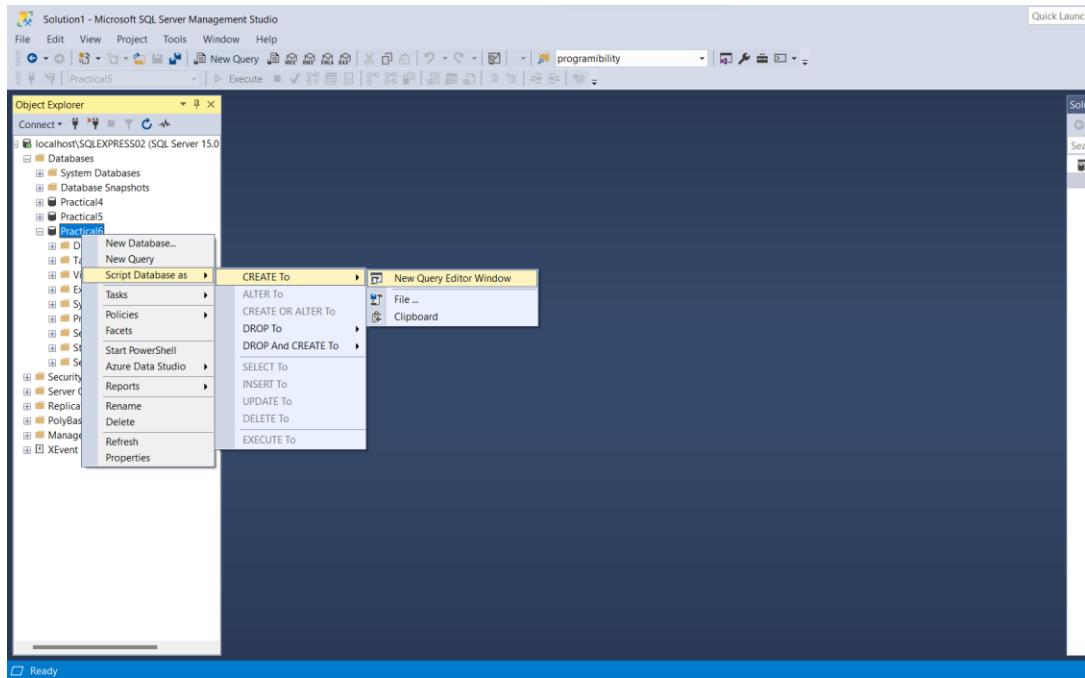


Steps:

1) Create database.



2) Script Database as > Create To > New Query Window



3) Create tables:

```
CREATE TABLE DIM_EMPLOYEE
```

```
(EMP_ID INT,  
EMP_NAME VARCHAR(25))
```

```
CREATE TABLE DIM_BRANCH
```

```
(BRANCH_ID INT,  
BRANCH_NAME VARCHAR(25))
```

```
CREATE TABLE DIM_PRODUCT
```

```
(PROD_ID INT,  
PROD_NAME VARCHAR(25),  
BRAND_ID INT)
```

```
CREATE TABLE DIM_BRAND
```

```
(BRAND_ID INT,  
BRAND_NAME VARCHAR(25))
```

```
CREATE TABLE DIM_CUSTOMER
```

```
(CUST_ID INT,  
CUST_NAME VARCHAR(25))
```

```
CREATE TABLE FACT_SHOP
```

```
(EMP_ID INT,  
BRANCH_ID INT,  
PROD_ID INT,
```

CUST_ID INT)

The screenshot shows the Microsoft SQL Server Management Studio interface. The title bar reads "SQLQuery3.sql - localhost\SQLEXPRESS02.Practical6 (MSI\syydz (67))* - Microsoft SQL Server Management Studio". The Object Explorer on the left shows the database structure, including the "Practical6" database and its tables: DIM_EMPLOYEE, DIM_BRANCH, DIM_PRODUCT, DIM_BRAND, DIM_CUSTOMER, and FACT_SHOP. The central pane displays the following T-SQL code:

```
CREATE TABLE DIM_EMPLOYEE
(EMP_ID INT,
EMP_NAME VARCHAR(25))

CREATE TABLE DIM_BRANCH
(BRANCH_ID INT,
BRANCH_NAME VARCHAR(25))

CREATE TABLE DIM_PRODUCT
(PROD_ID INT,
PROD_NAME VARCHAR(25),
BRAND_ID INT)

-----NORMALIZATION OF DIM_PRODUCT
CREATE TABLE DIM_BRAND
(BRAND_ID INT,
BRAND_NAME VARCHAR(25))

CREATE TABLE DIM_CUSTOMER
(CUST_ID INT,
CUST_NAME VARCHAR(25))

CREATE TABLE FACT_SHOP
(EMP_ID INT,
BRANCH_ID INT,
PROD_ID INT,
CUST_ID INT)
```

The status bar at the bottom indicates "Query executed successfully." and "Completion time: 2022-04-04T00:28:23.2534032+05:30".

4) Create New database diagram

The screenshot shows the Microsoft SQL Server Management Studio interface. The title bar reads "SQLQuery3.sql - localhost\SQLEXPRESS02.Practical6 (MSI\syydz (67))* - Microsoft SQL Server Management Studio". The Object Explorer on the left shows the database structure, including the "Practical6" database and its tables: DIM_EMPLOYEE, DIM_BRANCH, DIM_PRODUCT, DIM_BRAND, DIM_CUSTOMER, and FACT_SHOP. A context menu is open over the "Database Diagram" node under "Practical6", with the "New Database Diagram" option highlighted.

The central pane displays the same T-SQL code as the previous screenshot:

```
CREATE TABLE DIM_EMPLOYEE
(EMP_ID INT,
EMP_NAME VARCHAR(25))

CREATE TABLE DIM_BRANCH
(BRANCH_ID INT,
BRANCH_NAME VARCHAR(25))

CREATE TABLE DIM_PRODUCT
(PROD_ID INT,
PROD_NAME VARCHAR(25),
BRAND_ID INT)

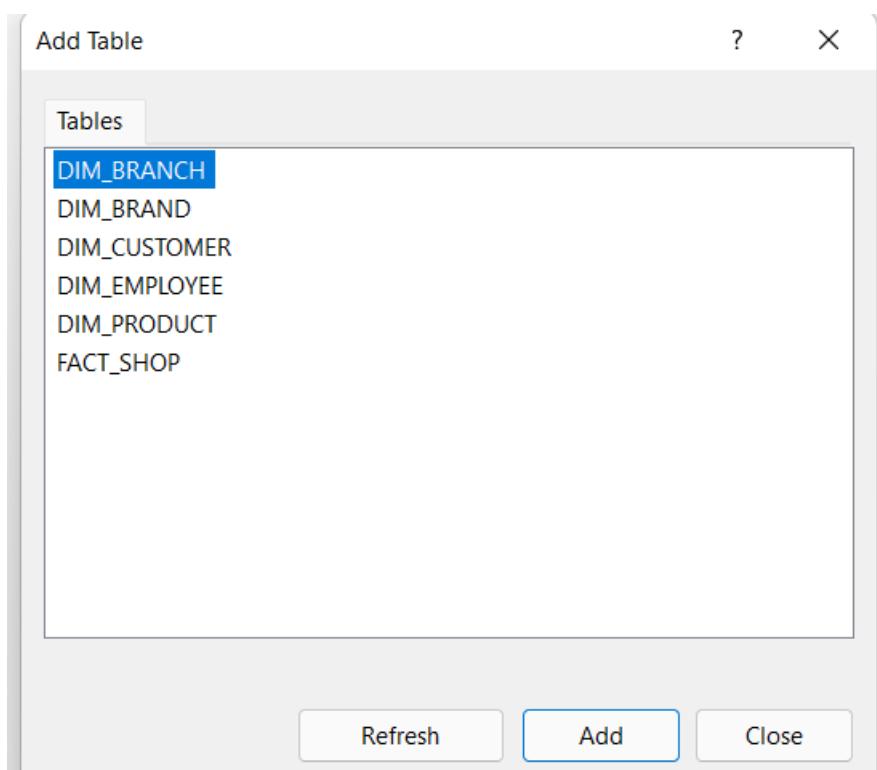
-----NORMALIZATION OF DIM_PRODUCT
CREATE TABLE DIM_BRAND
(BRAND_ID INT,
BRAND_NAME VARCHAR(25))

CREATE TABLE DIM_CUSTOMER
(CUST_ID INT,
CUST_NAME VARCHAR(25))

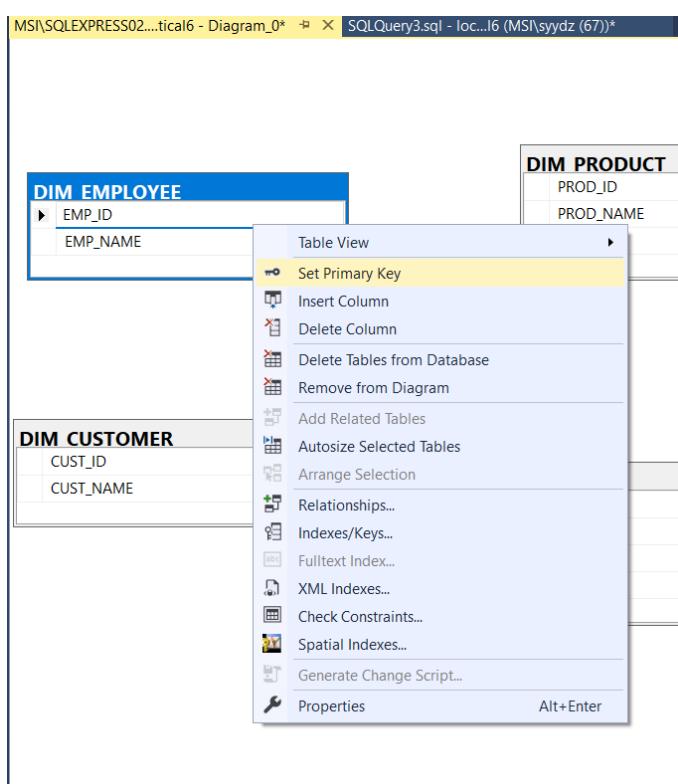
CREATE TABLE FACT_SHOP
(EMP_ID INT,
BRANCH_ID INT,
PROD_ID INT,
CUST_ID INT)
```

The status bar at the bottom indicates "Query executed successfully." and "Completion time: 2022-04-04T00:28:23.2534032+05:30".

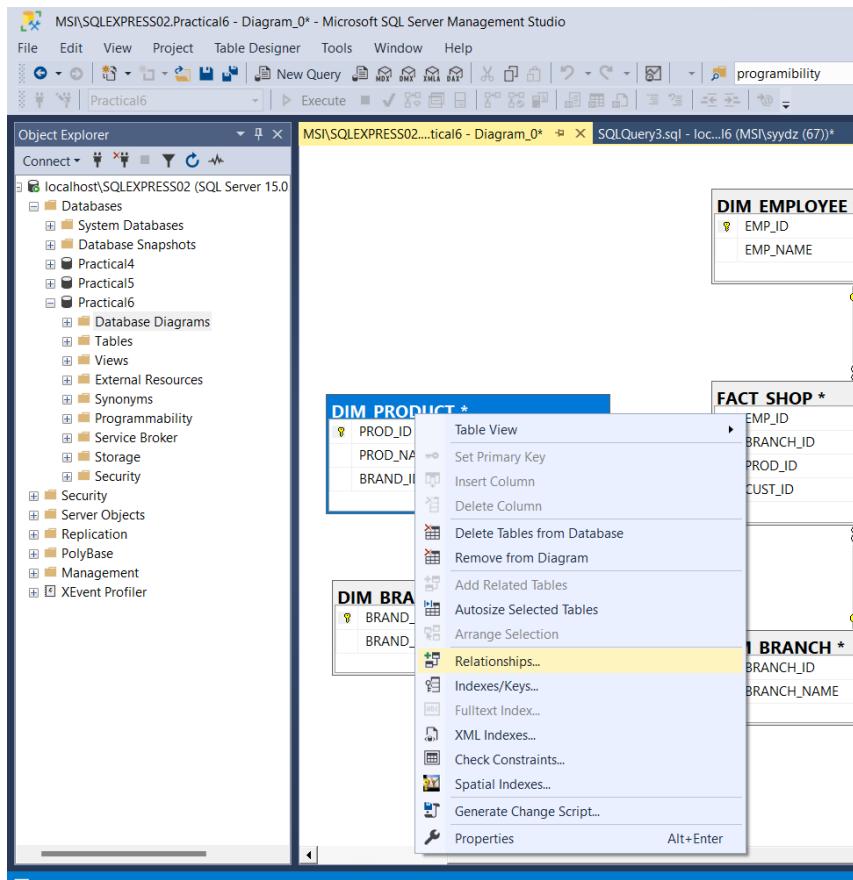
5) Add tables



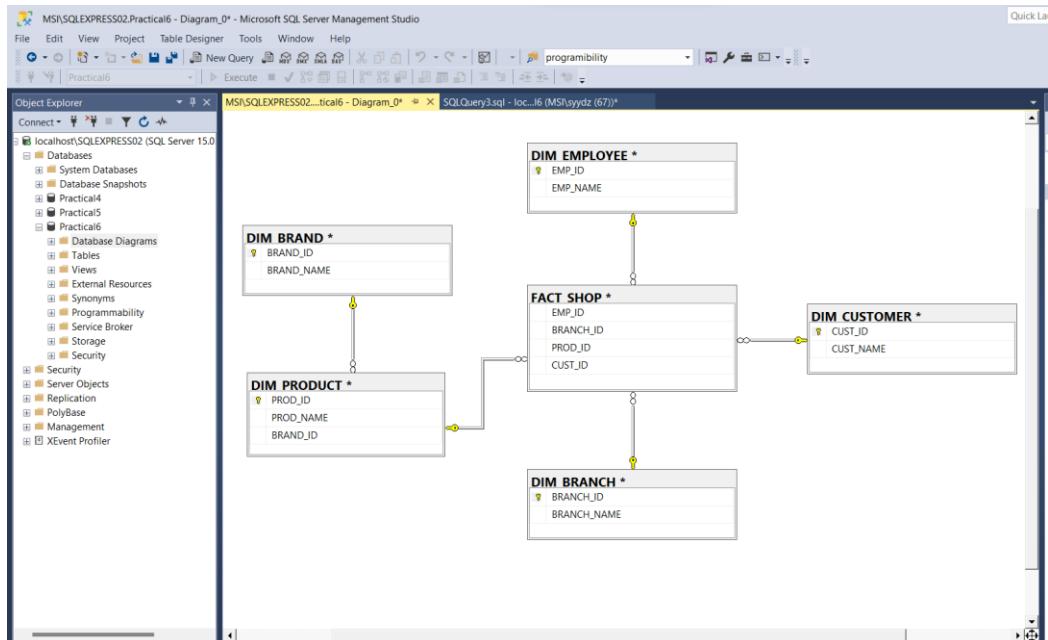
6) Set Primary key for all tables:



7) Add relationship of tables with fact shop.



8) Add relationship of product and brand .



Conclusion: Snowflake schema implemented successfully.

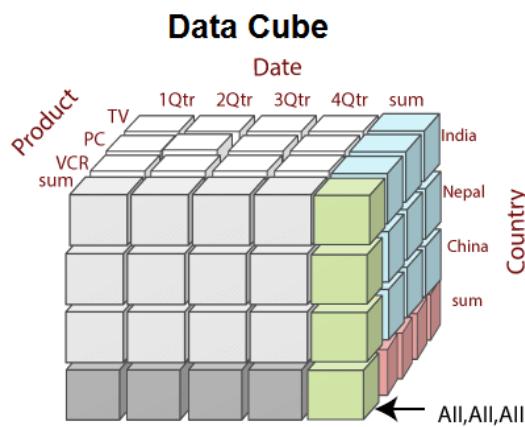
Practical 7

Aim: Create and uses excel pivot table report based on data cube operations.

Theory:

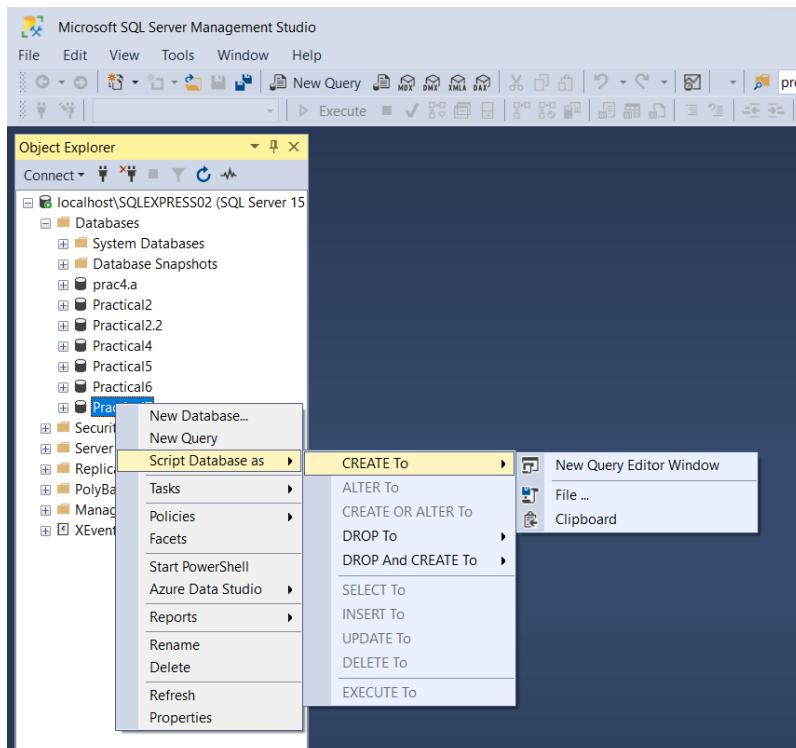
What is Data Cube?

When data is grouped or combined in multidimensional matrices called Data Cubes. The data cube method has a few alternative names or a few variants, such as "Multidimensional databases," "materialized views," and "OLAP (On-Line Analytical Processing)."



Steps:

- 1) Create new database



SQLQuery1.sql - localhost\SQLEXPRESS02.Practical7 (MSI\syydz (60)) - Microsoft SQL Server Management Studio

File Edit View Query Project Tools Window Help

localhost\SQLEXPRESS02 (SQL Server 15)

Object Explorer

SQLQuery1.sql - loc...l7 (MSI\syydz (60))

```
CREATE TABLE Purchase
(
    PROD_ID INT ,
    PROD_NAME VARCHAR(25),
    REGION VARCHAR(25),
    UNIT INT,
    AMOUNT INT
)
```

100 %

Messages

Commands completed successfully.

Completion time: 2022-04-17T10:26:53.0984686+08:00

SQLQuery1.sql - localhost\SQLEXPRESS02.Practical7 (MSI\syydz (60)) - Microsoft SQL Server Management Studio

File Edit View Query Project Tools Window Help

localhost\SQLEXPRESS02 (SQL Server 15)

Object Explorer

SQLQuery1.sql - loc...l7 (MSI\syydz (60))

```
CREATE TABLE Purchase
(
    PROD_ID INT ,
    PROD_NAME VARCHAR(25),
    REGION VARCHAR(25),
    UNIT INT,
    AMOUNT INT
)
```

localhost\SQLEXPRESS02 (SQL Server 15)

Databases

- System Databases
- Database Snapshots
- prac4.a
- Practical2
- Practical2.2
- Practical4
- Practical5
- Practical6
- Practical7
 - Database Diagrams
 - Tables
 - System Tables
 - FileTables
 - External
 - Graph Tables
 - dbo.Purchase
 - Views
 - External Resources
 - Synonyms
 - Programmability
 - Service Broker
 - Storage
 - Security

Security

Server Objects

Replication

PolyBase

Tables

System Tables

FileTables

External

Graph Tables

dbo.Purchase

- Columns
- Keys
- Constraints
- Triggers
- Indexes
- Statistics

Views

External Resources

Synonyms

Programmability

Service Broker

Storage

Security

Script Table as

CREATE To

ALTER To

CREATE OR ALTER To

DROP To

DROP And CREATE To

SELECT To

INSERT To

UPDATE To

DELETE To

EXECUTE To

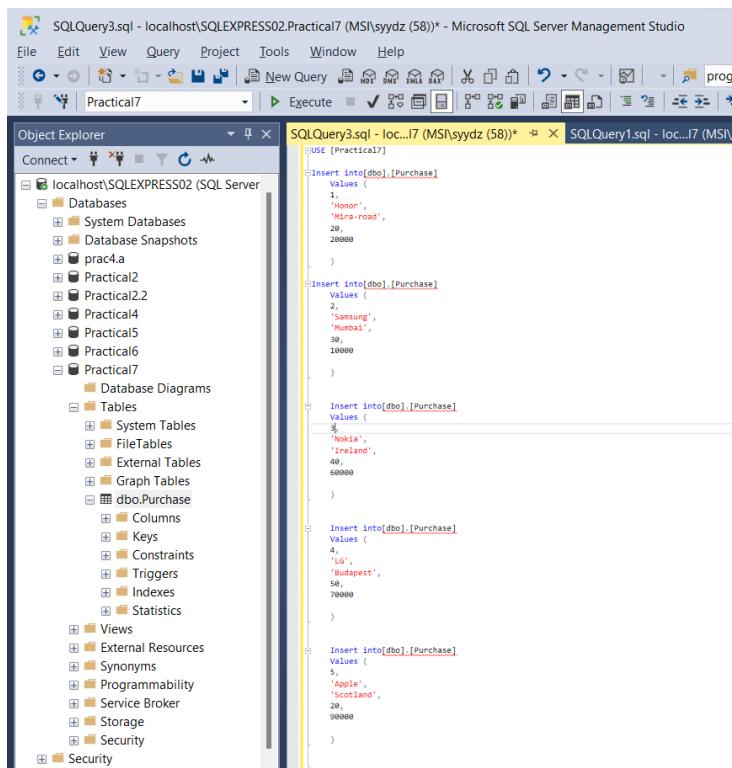
New Query Editor Window

File ...

Clipboard

successfully.

2) Insert data in table



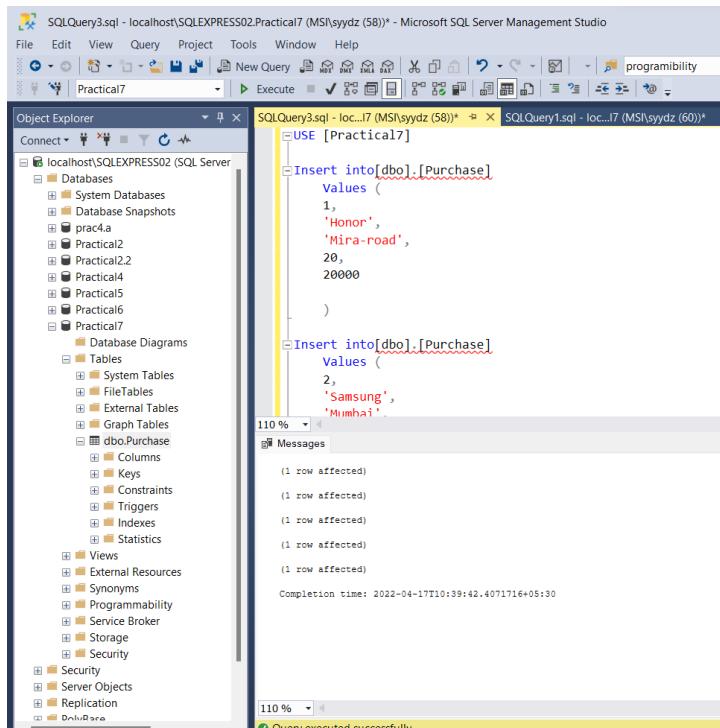
```
USE [Practical17]
Insert into[dbo].[Purchase]
Values (
1,
'Honor',
'Mira-road',
20,
20000
)

Insert into[dbo].[Purchase]
Values (
2,
'Samsung',
'Mumbai',
30,
10000
)

Insert into[dbo].[Purchase]
Values (
3,
'Nokia',
'Ireland',
40,
60000
)

Insert into[dbo].[Purchase]
Values (
4,
'LG',
'Budapest',
50,
70000
)

Insert into[dbo].[Purchase]
Values (
5,
'Apple',
'Scotland',
20,
90000
)
```



```
USE [Practical17]
Insert into[dbo].[Purchase]
Values (
1,
'Honor',
'Mira-road',
20,
20000
)

Insert into[dbo].[Purchase]
Values (
2,
'Samsung',
'Mumbai',
30,
10000
)
```

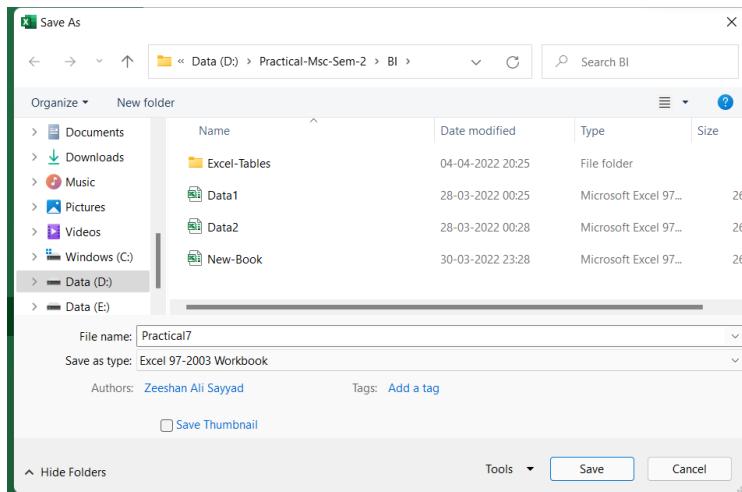
Messages

(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)

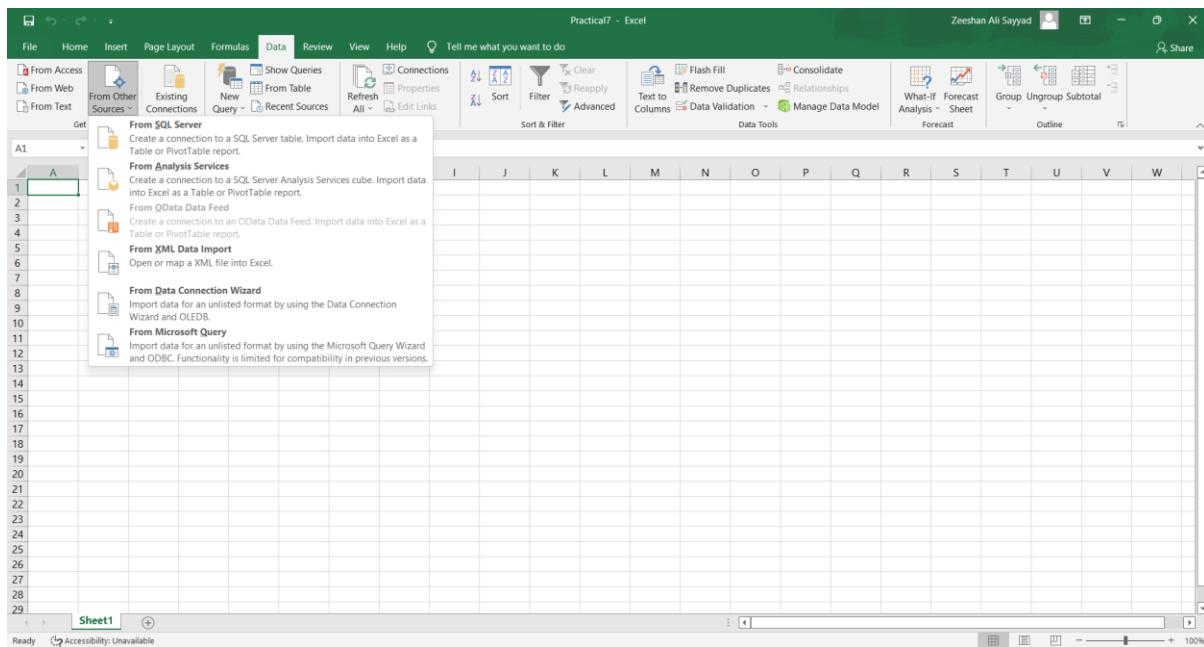
Completion time: 2022-04-17T10:39:42.4071716+05:30

Query executed successfully.

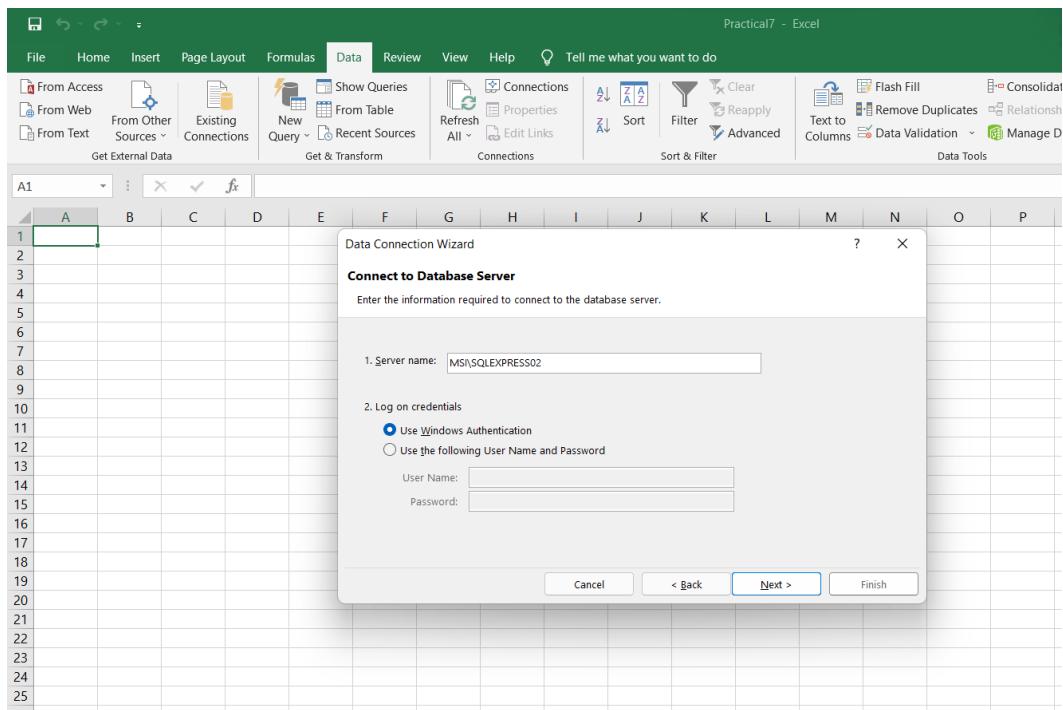
3) Create a new Excel File



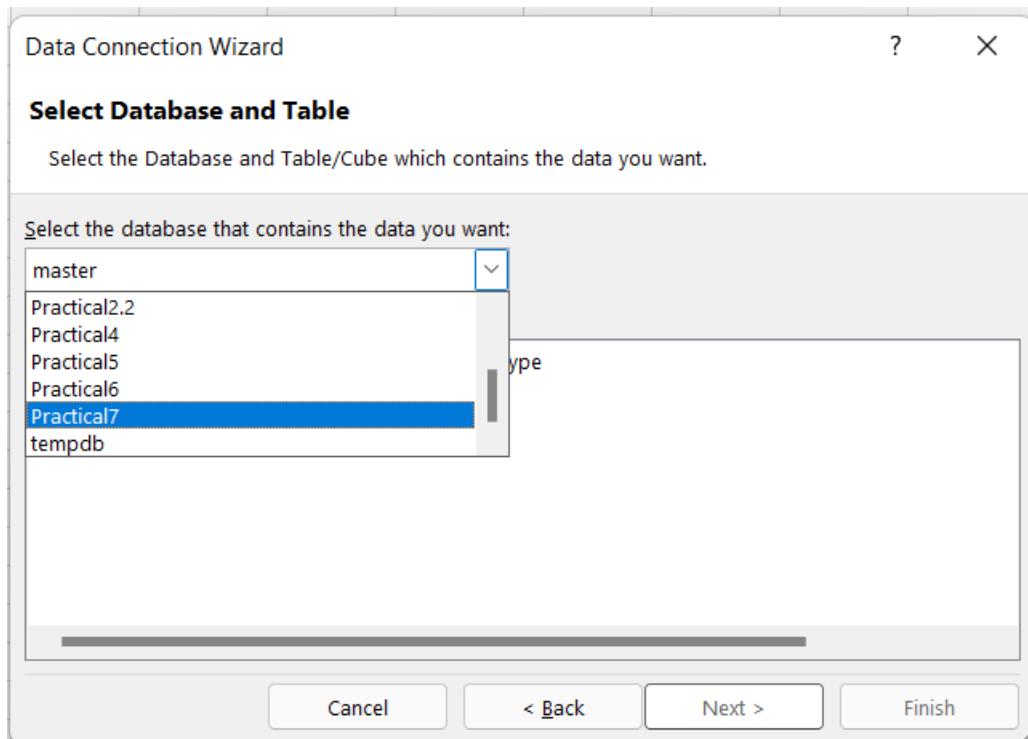
4) Go to Data > From Other Sources> Click on From SQL Server



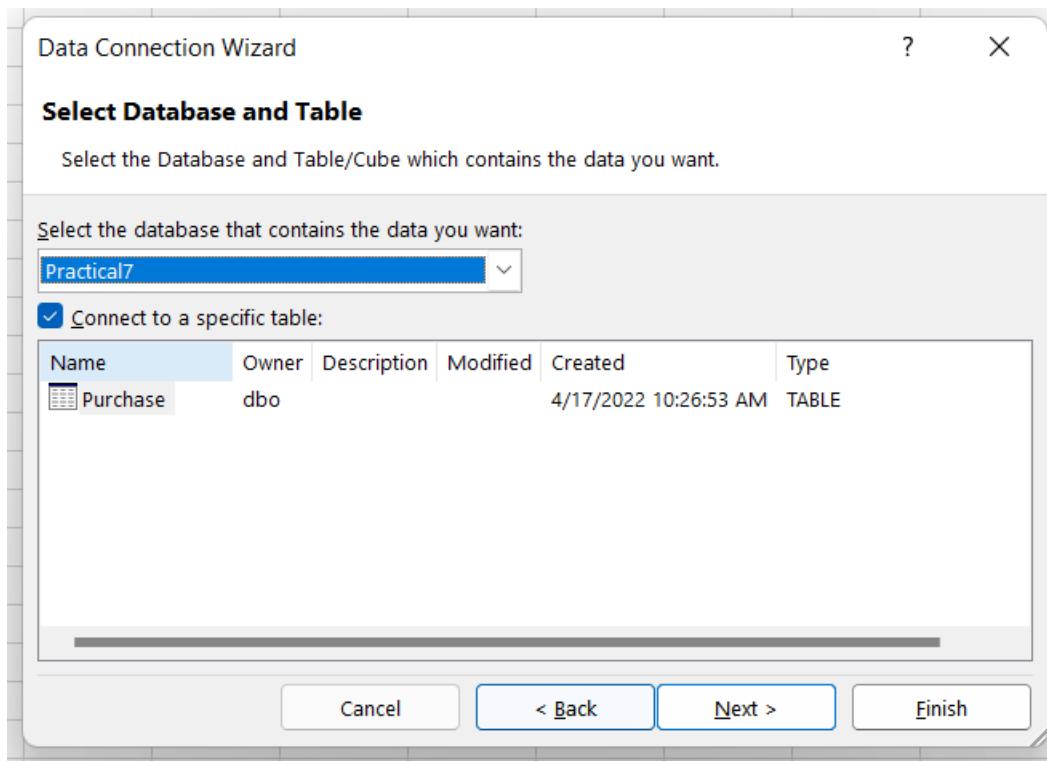
5) Connect to database using server name.



6) Select database and table we want to connect to.



7) Click on Next.



8) Select PivotTable report

The screenshot shows an Excel spreadsheet titled 'Practical7 - Excel'. The 'Data' tab is selected in the ribbon. An 'Import Data' dialog box is open, prompting the user to select how to view the data: 'Table' (radio button), 'PivotTable Report' (radio button, selected), 'PivotChart', or 'Only Create Connection'. It also asks where to put the data: 'Existing worksheet' (radio button) with the range '\$A\$1' selected, or 'New worksheet'. There is a checkbox for 'Add this data to the Data Model'. At the bottom are 'Properties...', 'OK', and 'Cancel' buttons.

9) Arrange fields PivotTable Fields respectively.

| | | | | | Sum of AMOUNT | Total Sum of AMOUNT | Total Sum of UNIT |
|----|--------------------|---------------|--------------|--------------|---------------|---------------------|-------------------|
| | | | | | 1 | 2 | 3 |
| 6 | Row Labels | Column Labels | | | Sum of UNIT | | |
| 7 | Apple | | | | 90000 | 20 | 90000 |
| 8 | Scotland | | | | 90000 | 20 | 90000 |
| 9 | Honor | 20000 | | | | 20 | 20000 |
| 10 | Mira-road | 20000 | | | | 20 | 20000 |
| 11 | LG | 70000 | | | | 50 | 70000 |
| 12 | Budapest | 70000 | | | | 50 | 70000 |
| 13 | Nokia | 60000 | | | | 40 | 60000 |
| 14 | Ireland | 60000 | | | | 40 | 60000 |
| 15 | Samsung | 10000 | | | | 30 | 10000 |
| 16 | Mumbai | 10000 | | | | 30 | 10000 |
| 17 | Grand Total | 20000 | 10000 | 60000 | 70000 | 90000 | 250000 |
| | | | | | | 20 | 160 |

Conclusion: Successfully implemented pivot table report based on data cube operations.

Practical 8

Aim: Prerequisites for Database applications.

Theory: A relational database organizes data into tables which can be linked—or *related*—based on data common to each. This capability enables you to retrieve an entirely new table from data in one or more tables with a single query. It also allows you and your business to better understand the relationships among all available data and gain new insights for making better decisions or identifying new opportunities.

Queries:

Creating employee table:

```
SQL>create table employee(  
    empid number(3) primary key,  
    empname varchar2(25),  
    age number(3));
```

Creating client table:

```
SQL>create table client(  
    clientid number(3) primary key,  
    clientname varchar(25),  
    location varchar2(20));
```

Creating project table:

```
SQL>create table project(  
    projid number(5) primary key,  
    empid number(3),  
    projname varchar2(25),  
    clientid number(3));
```

Inserting records in table employee:

```
SQL>insert into employee values(1,'Vardhan',22);  
SQL>insert into employee values(2,'Khushi',30);  
SQL>insert into employee values(3,'Simran',40);  
SQL>insert into employee values(4,'Chetan',20);  
SQL>insert into employee values(5,'Swati',27);  
SQL>insert into employee values(10,'Vedant',33);  
SQL>insert into employee values(15,'Gauri',35);  
SQL>insert into employee values(16,'ABC',36);  
SQL>insert into employee values(17,'PQR',37);
```

Inserting records in table client:

```
SQL>insert into client values(33,'will','mumbai');  
SQL>insert into client values(34,'jones','bangalore');  
SQL>insert into client values(35,'deep','kolkata');
```

```
SQL>insert into client values(36,'smith','delhi');
SQL>insert into client values(37,'marin','mumbai');
SQL>insert into client values(38,'jason','mumbai');
```

Inserting records in table project:

```
SQL>insert into project values(111,1,'abc',33);
SQL>insert into project values(222,2,'def',34);
SQL>insert into project values(333,3,'xyz',35);
SQL>insert into project values(444,4,'pqr',36);
SQL>insert into project values(555,5,'mnq',37);
SQL>insert into project values(666,6,'efg',38);
SQL>insert into project values(777,7,'lmn',39);
SQL>insert into project values(888,8,'ppd',40);
```

```
SQL> desc employee;
Name          Null?    Type
-----        -----   -----
EMPID         NOT NULL NUMBER(3)
EMPNAME       VARCHAR2(25)
AGE           NUMBER(3)
```

```
SQL> desc client;
Name          Null?    Type
-----        -----   -----
CLIENTID      NOT NULL NUMBER(3)
CLIENTNAME    VARCHAR2(25)
LOCATION      VARCHAR2(20)
```

```
SQL> desc project;
Name          Null?    Type
-----        -----   -----
PROJID        NOT NULL NUMBER(5)
EMPID         NUMBER(3)
PROJNAME      VARCHAR2(25)
CLIENTID      NUMBER(3)
```

```
SQL> select * from employee;
EMPID  EMPNAME          AGE
-----  -----            -----
1      Vardhan           22
2      Khushi            30
3      Simran            40
4      Chetan            20
5      Swati             27
```

```
SQL> select * from client;
CLIENTID  CLIENTNAME          LOCATION
-----    -----            -----
33        will                mumbai
34        jones               bangalore
35        deep                kolkata
36        smith               delhi
37        marin               mumbai
38        jason               mumbai
```

```
SQL> insert into project values(111,1,'abc',33);
1 row created.
```

```

SQL> select * from project;

  PROJID    EMPID PROJNAME          CLIENTID
-----  -----  -----
    111        1 abc
    222        2 def
    333        3 xyz
    444        4 pqr
    555        5 mnq
    666        6 efg
    777        7 lmn
    888        8 ppd

8 rows selected.

```

```

SQL> select employee.empid,employee.empname,project.clientid
  2  from employee
  3  INNER JOIN
  4  project
  5  ON employee.empid=project.empid;

  EMPID EMPNAME          CLIENTID
-----  -----
    1 Vardhan
    2 Khushi
    3 Simran
    4 Chetan
    5 Swati

```

```

SQL> select employee.empid,employee.empname,project.clientid
  2  from employee
  3  FULL JOIN
  4  project
  5  ON employee.empid=project.empid;

  EMPID EMPNAME          CLIENTID
-----  -----
    1 Vardhan
    2 Khushi
    3 Simran
    4 Chetan
    5 Swati
    6
    7
    8
    9
    10

```

```

SQL> select employee.empid, employee.empname,project.clientid
  2  from employee
  3  LEFT JOIN
  4  project
  5  ON employee.empid=project.empid;

  EMPID EMPNAME          CLIENTID
-----  -----
    1 Vardhan
    2 Khushi
    3 Simran
    4 Chetan
    5 Swati

```

```
SQL> select employee.empid, employee.empname,project.clientid
  2  from employee
  3  RIGHT JOIN
  4  project
  5  ON employee.empid=project.empid;

      EMPID  EMPNAME          CLIENTID
-----  -----  -----
        1  Vardhan            33
        2  Khushi              34
        3  Simran              35
        4  Chetan              36
        5  Swati               37
                                40
                                38
                                39

8 rows selected.
```

Transactions

```
SQL> set transaction read write;
Transaction set.

SQL> UPDATE EMPLOYEE
  2  SET EMPID = 201
  3  WHERE EMPID = 1;

1 row updated.

SQL> commit;
Commit complete.

SQL> insert into employee values(16,'ABC',36);
1 row created.

SQL> insert into employee values(17,'PQR',37);
1 row created.
```

Savepoint

```
SQL> savepoint s1;
Savepoint created.

SQL> insert into employee values(6,'ABC',20);
1 row created.
```

Rollback

```
SQL> rollback to s1;  
Rollback complete.  
  
SQL> select * from employee;  
EMPID EMPNAME AGE  
-----  
201 Vardhan 22  
2 Khushi 30  
3 Simran 40  
4 Chetan 20  
5 Swati 27
```

Wild cards

```
SQL> select * from employee  
2 where empname like '%an';  
  
EMPID EMPNAME AGE  
-----  
201 Vardhan 22  
3 Simran 40  
4 Chetan 20  
  
SQL> select * from employee  
2 where empname like 'Khu%';  
  
EMPID EMPNAME AGE  
-----  
2 Khushi 30  
  
SQL> select * from employee  
2 where empname like '%ar%';  
  
EMPID EMPNAME AGE  
-----  
201 Vardhan 22
```

Distinct

```
SQL> insert into employee values(6, 'ABC', 20);  
1 row created.  
  
SQL> insert into employee values(7, 'ABC', 22);  
1 row created.  
  
SQL> select distinct age from employee;  
AGE  
----  
22  
30  
20  
40  
27
```

Conclusion: Prerequisites for Database applications completed successfully.